Essentials of HDCP 2.2 Authentication & Encryption Protocols
for HDMI and DisplayPort

Neal Kendall – Marketing Manager
Teledyne LeCroy quantumdata Product Family
neal.kendall@teledyne.com

TELEDYNE LECROY
Everywhereyoulook™

quantumdata™

HDCP 2.2 Webinar Agenda

- What is HDCP 2.2?
- Overview of Modern Cryptography
- HDCP 2.2 Authentication & Key Exchange, Pairing, Locality Check
- HDCP 2.2 Encryption
- HDCP 2.2 Compliance Testing

- Please Check out our other “Essentials of” Webinars:
  - Essentials of DisplayPort Protocols
  - Essentials of HDCP 2.2 Protocols
  - Essentials of HDMI Fixed Rate Link (FRL) Protocols
  - Essentials of DisplayPort Display Stream (DSC) Protocols

HDCP Overview

TELEDYNE LECROY
Everywhereyoulook™

Why HDCP?

- Remember the old days when you could copy movies on a VCR through an analog video interface cable?

What is HDCP?

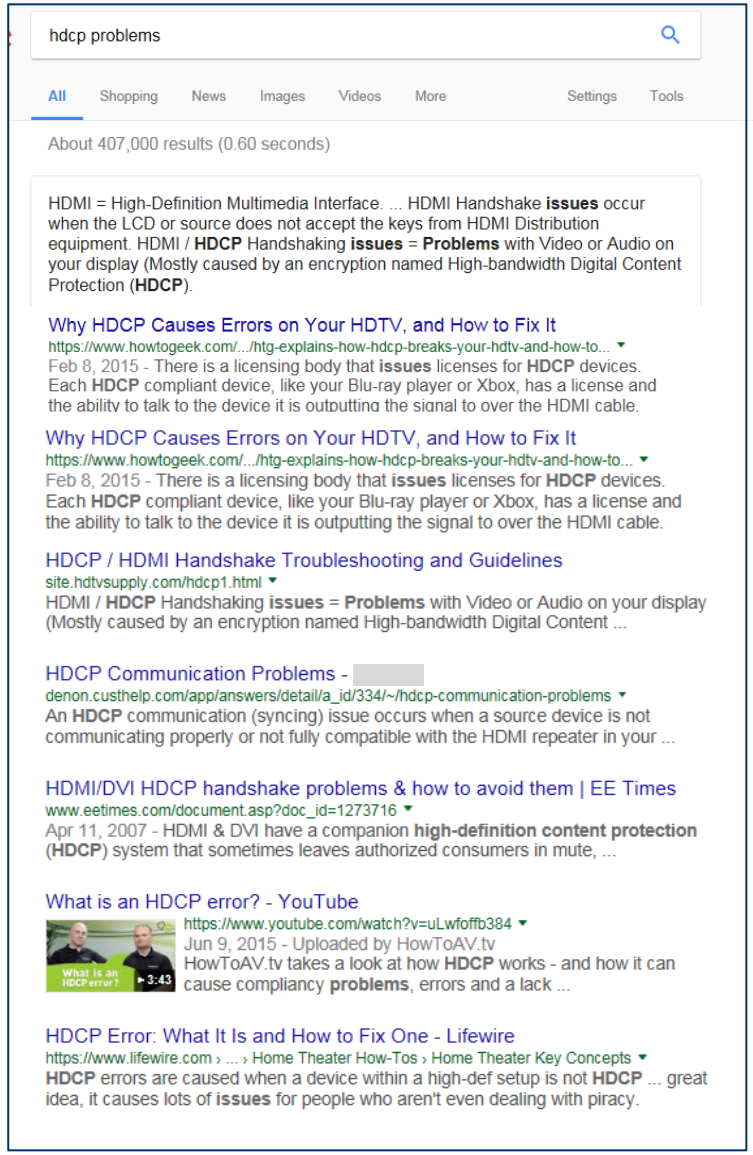
- High-Bandwidth Digital Content Protection (HDCP) - A form of digital copy protection or “Digital Rights Management.”
- Developed by Intel Corporation in 2003.
- Licensing governed by Digital Content Protection, LLC (DCP).
- From DCP website:

“Digital Content Protection LLC (DCP) is an organization that licenses technologies for protecting premium commercial entertainment content. High-bandwidth Digital Content Protection (HDCP) is a specification developed by Intel Corporation to protect digital entertainment content across digital interfaces.”

- Provides a protection mechanism over the physical link, e.g. HDMI and DisplayPort.
- Control access—modification and distribution--of proprietary copyrighted material, i.e. television and motion picture content.
- HDCP 2.2 uses modern cryptography mechanisms.

HDCP Interoperability Problems

- Many people are aware of HDCP only because of inter-operability problems while connecting DisplayPort or HDMI or devices.
- Some experts have said that HDCP is the most difficult thing to get right about HDMI and DisplayPort.



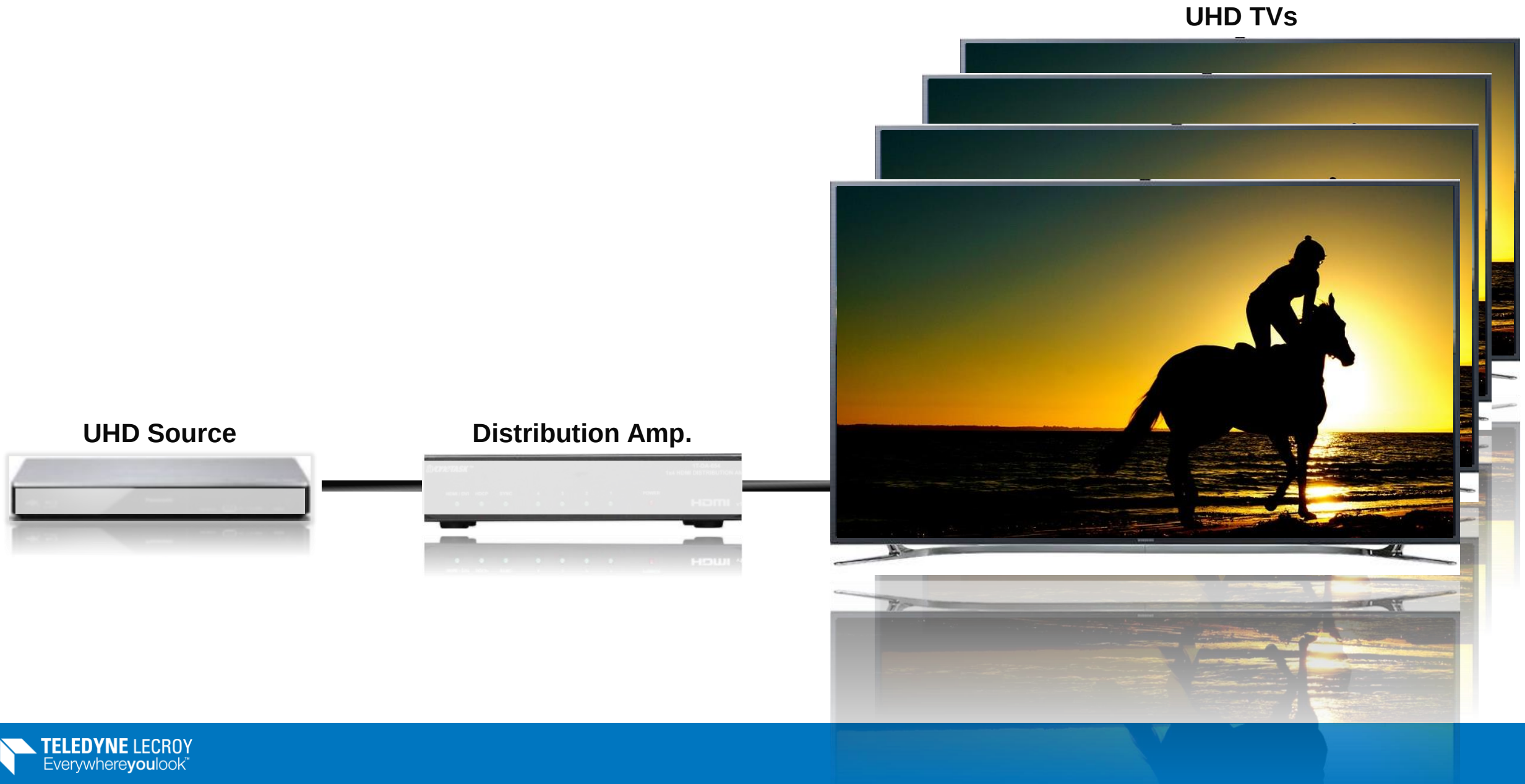
What are the Functions of HDCP?

- HDCP supports three (3) primary functions:
  - **Authentication** – A protocol exchange initiated by the HDCP Transmitter to verify that a Receiver is licensed to receive HDCP content.
  - **Encryption** – The process by which HDCP encodes protected content such that only authorized devices can use it.
  - **Revocation** (Renewability) – Is a provision allowing HDCP to revoke the license of a receiver based on non-compliant behavior.

Basic HDCP 2.2 System - HDMI HDCP Transaction Channel - DDC

DisplayPort HDCP 2.2 Transaction Channel – Aux Channel

HDCP System with Distribution Device



Why HDCP 2.2?

- HDCP 1.x had some vulnerabilities.
- HDCP 2.2, not a continuation of HDCP 1.x.
- Robust, standards based link protection scheme.
  - RSA cryptography for key exchange during for authentication.
  - Advanced Encryption Standard (AES) for encryption.
- New features:
  - “Pairing” to expedite authentication on subsequent authentications.
  - “Locality Check” to ensure the receiver is relatively close.

Overview of Modern Cryptography

TELEDYNE LECROY
Everywhereyoulook™

Cryptography

- **Definition**: Cryptography is the practice and study of techniques for securing communication in the presence of third party adversaries.

- Cryptographic functions include:
  - Hashing

  - Random Number Generation

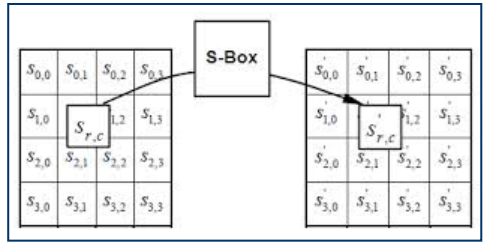
  - Encryption

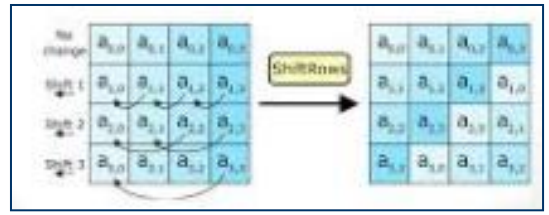
  - Digital Certificates and Digital Signatures

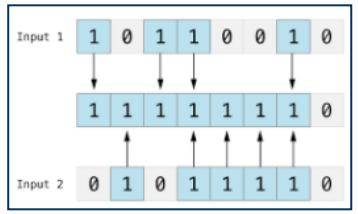
Cryptography

- Modern cryptography uses mathematical algorithms to transform information.
- There are three primitives to achieve this:
  - Substitution – Change the characters in a message.
  - Transposition – Rearrange the characters in a message.
  - Bitwise operation – XOR, AND, NOT, OR.

  Note: These processes are often repeated multiple times.
- HDCP uses cryptography to encrypt audio and video content.



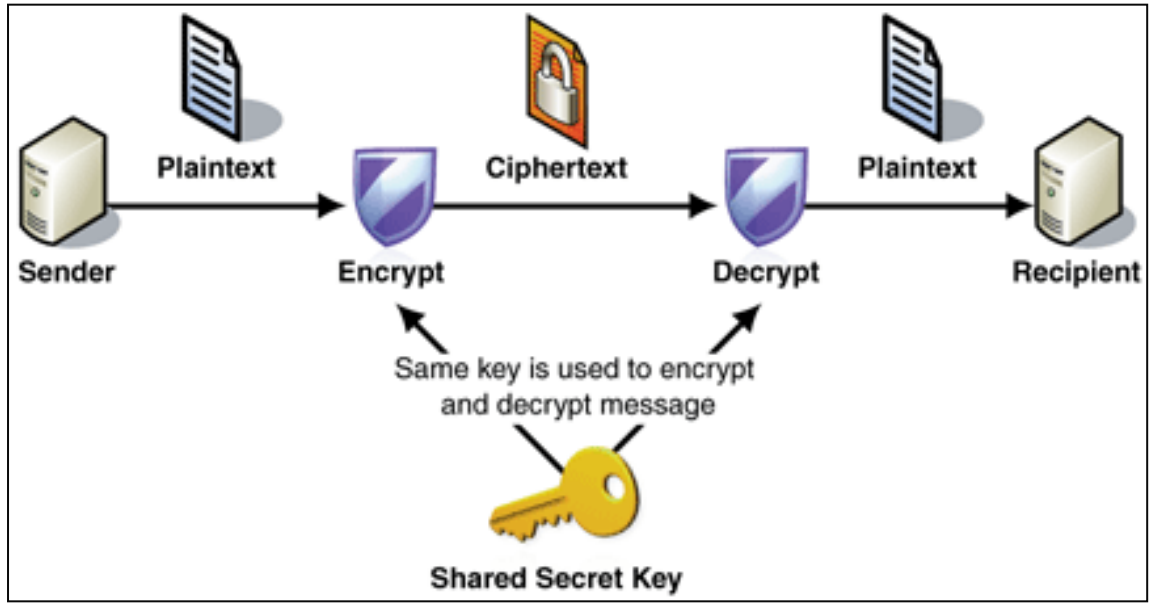




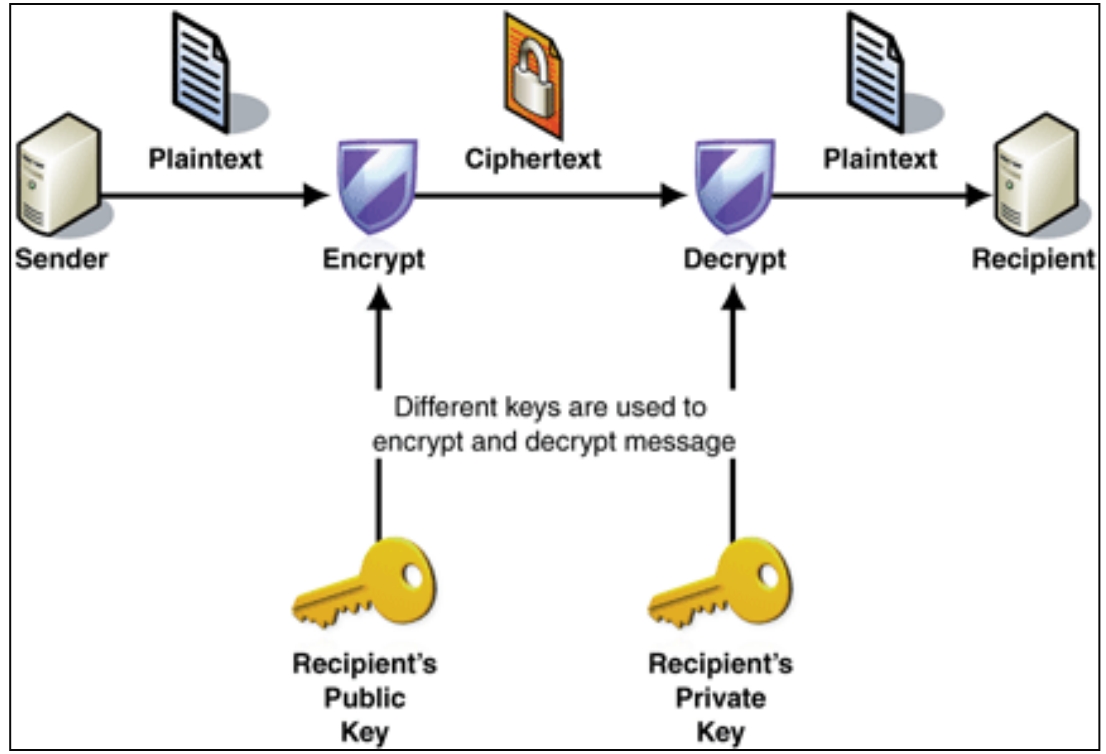
Types of Modern Cryptography

- Cryptographic systems can be categories by they keys they use:
  - Symmetric-Key (Private-Key) Encryption. (AES)
  - Asymmetric-Key (Public-Key) Encryption. (RSA)

Symmetric (Private) Key Encryption (AES)



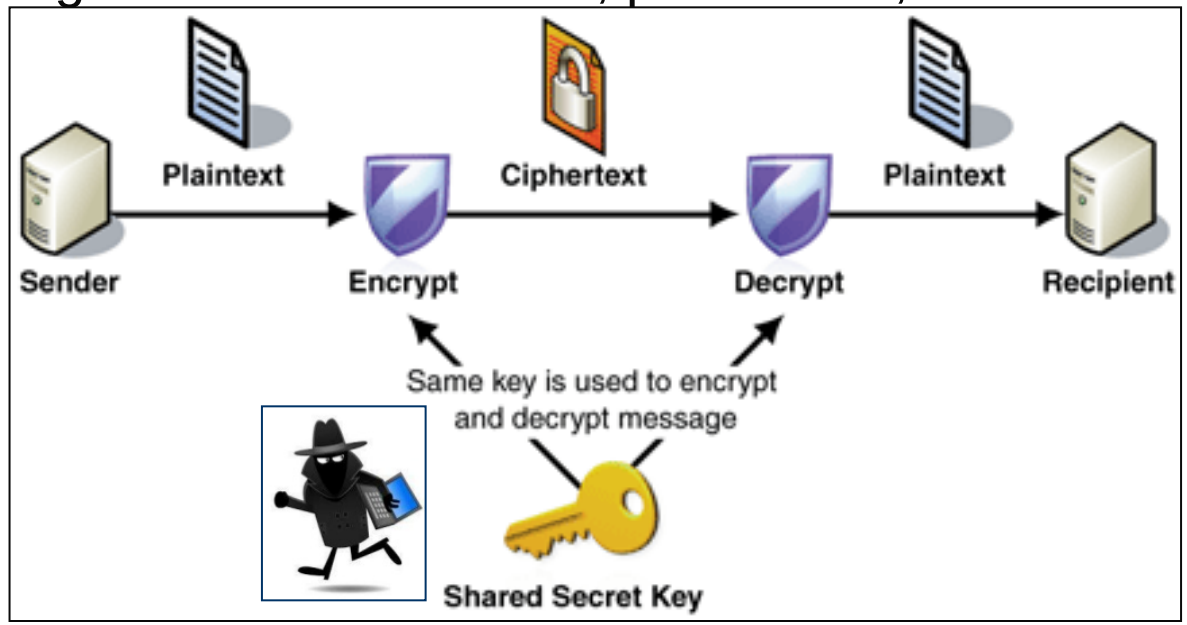
Asymmetric (Public) Key Encryption (RSA)



Types of Cryptography - Symmetric (Private) Key Encryption (AES)

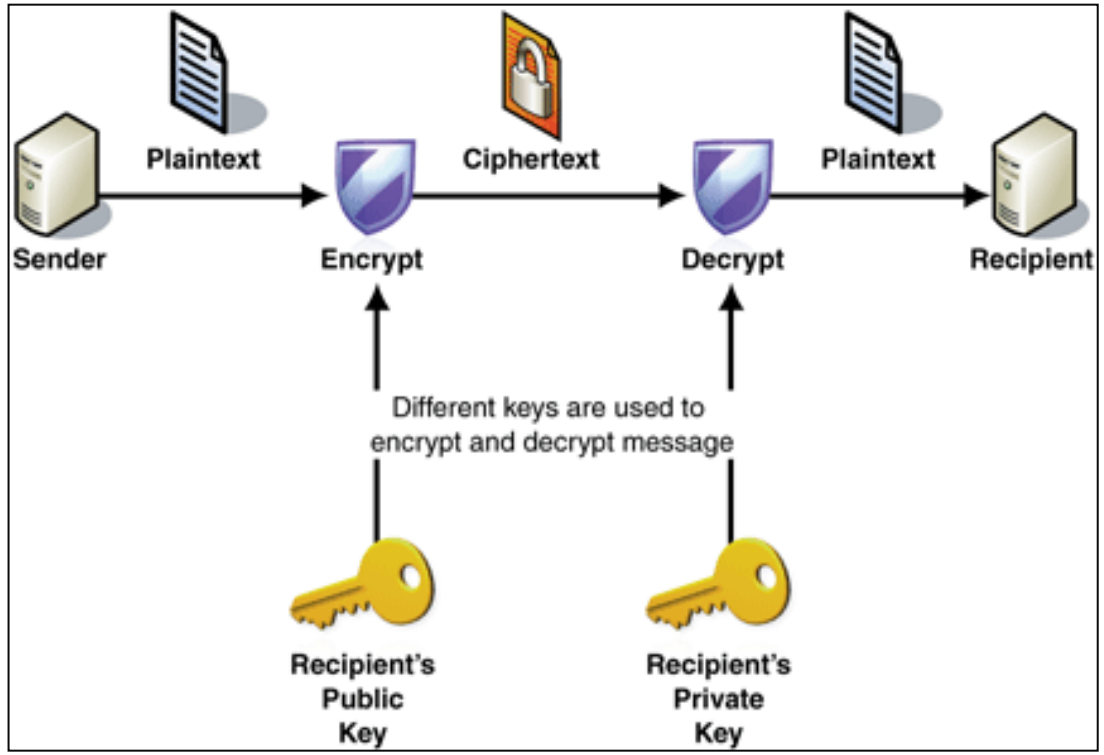
- **Definition**: A Cryptographic process using a ***shared, secret, private*** key for encrypting & decrypting information between two parties.
- The commonly used Data Encryption Standard (DES) and the newer, more robust Advanced Encryption Standard (AES) (used in HDCP 2.2) are forms of symmetrical block cipher techniques.
- Symmetric Key encryption is ***much faster*** than Asymmetric Key encryption.
- Used primarily when have to store data in a single place "*data at rest*."
  Examples are: storing data in a database and storing credit card numbers, passwords, etc.
- Main **drawbacks**:
  - Secret key needs to be ***stored*** securely.
  - Requires a *pre-agreed* upon secret key, or…
  - …a *secure channel* to exchange the secret key.
  - Requires *separate keys for each authentication* party. There is an explosion in the number of keys required.

  Note: In Consumer Electronic devices there is a need to exchange information with non-familiar devices



Cryptography – Asymmetric (Public) Key Encryption (RSA)

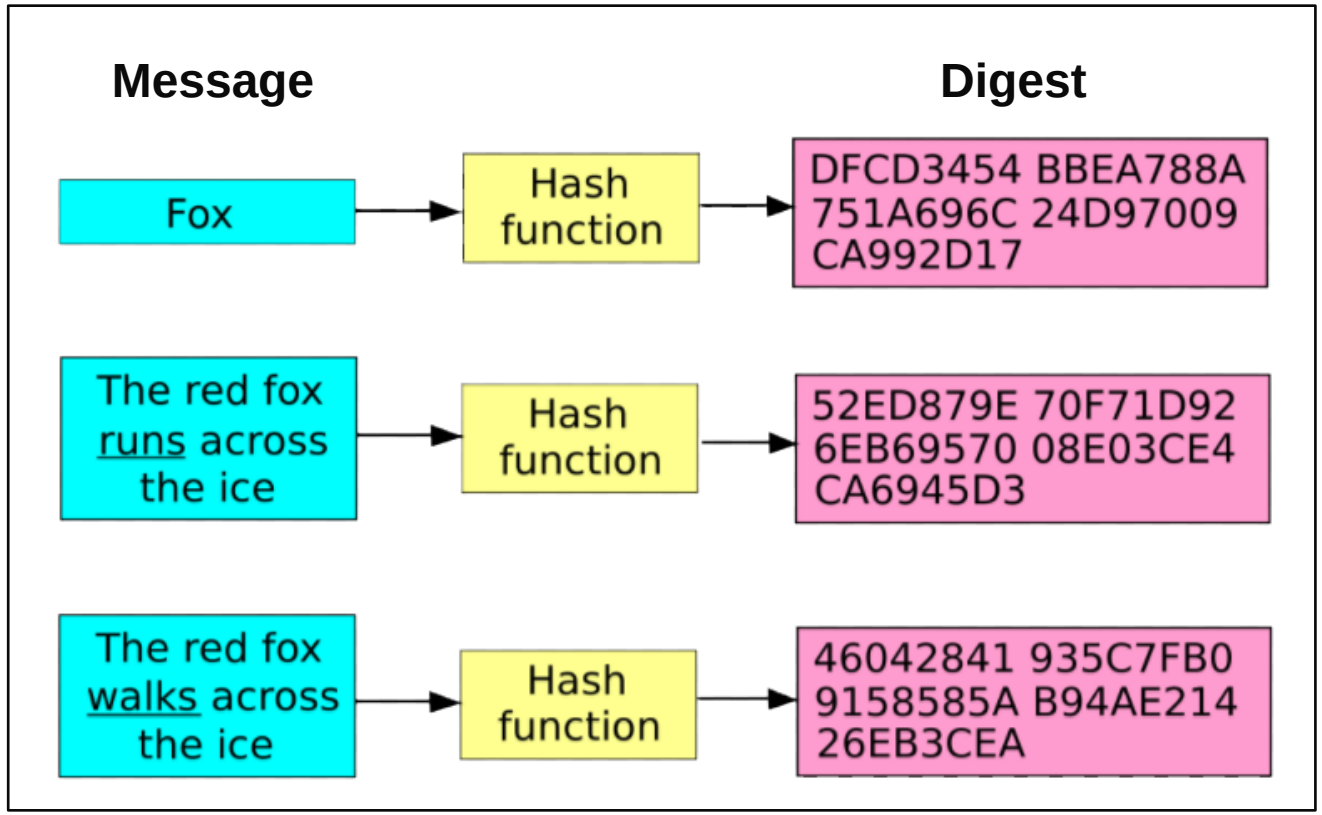
- A cryptographic system that uses pairs of keys: a **Public Key** which may be disseminated widely, and a **Private Key** which is known only to the owner—e.g. The Receiver.
- Invented to address the need for exchanging a secret key over an unsecured link.
- Used when there are separate device endpoints, e.g. web browsers, VPNs, secure shell, secure FTP.
- RSA is one of the first Asymmetric Key encryption systems and is widely used and used in HDCP 2.2.
- Because of the computational complexity of Asymmetric Key encryption, it is ***very slow***.
- Used only for small blocks of data.
- Many modern systems are a ***hybrid*** between symmetric and asymmetric key encryption.
- ***Often used to transfer a key that will be used for symmetric key encryption*** (e.g. “Master Key” in HDCP 2.2).



Cryptography
Hash Functions

Cryptographic Hash Functions – What Are They?

- **Definition**: A cryptographic hash function is a mathematical algorithm that can be used to map data of ***arbitrary size*** to data of a ***fixed size***.
- They have been described as the “work horses” of modern cryptography.
- Cryptographic hash functions are said to be “***one-way functions***” because they cannot easily be reversed. The only way to recreate the input data (“message”) from an ideal cryptographic hash function's output (“Digest”) is to attempt a brute-force search of possible inputs.
- **Wikipedia**: “SHA-2 is a set of cryptographic hash functions designed by the National Security Agency (NSA).”
- SHA-256 is a common type of Secure Hash Algorithm.
  - Used extensively in HDCP 2.2.
  - Produces a “Digest” of 256 bits regardless of the size of the message input.

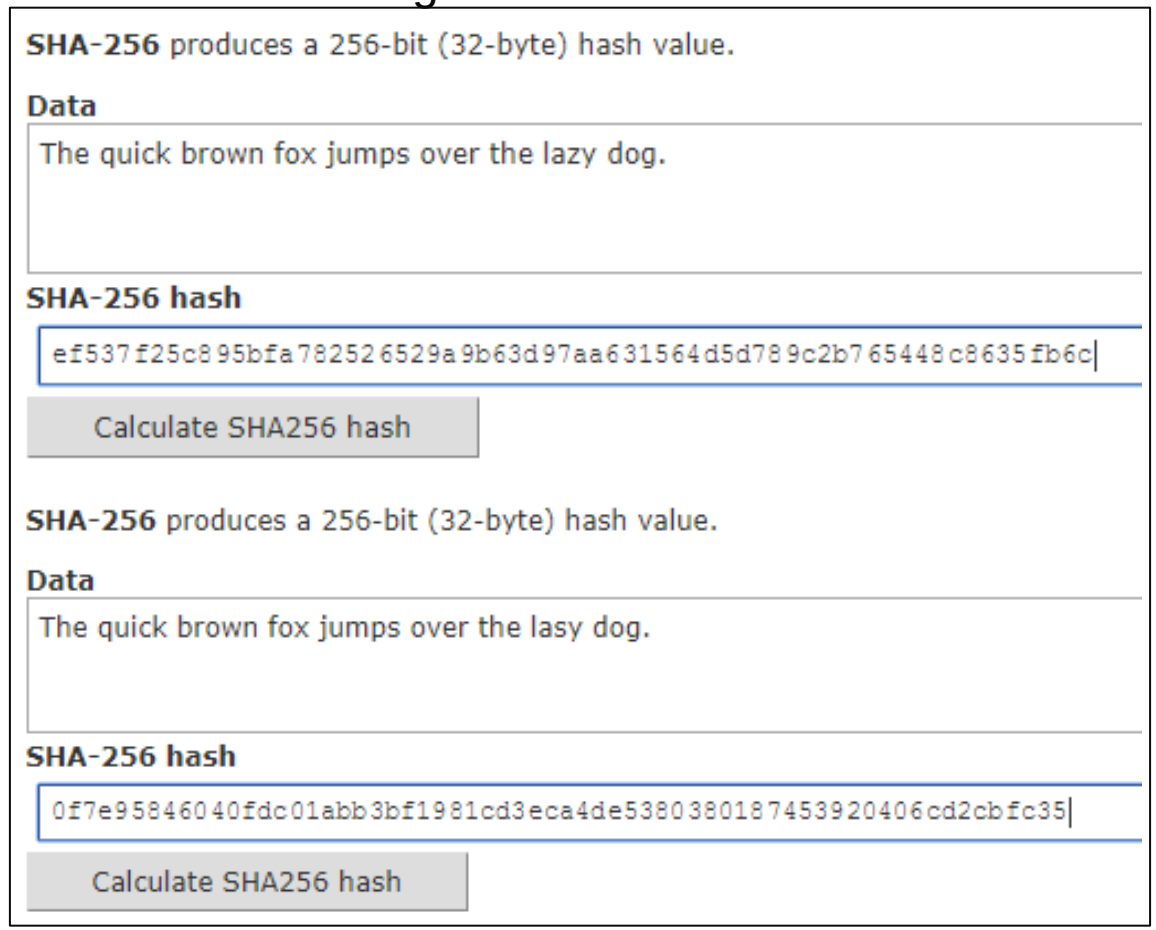


Cryptographic Hash Functions – What Are They?

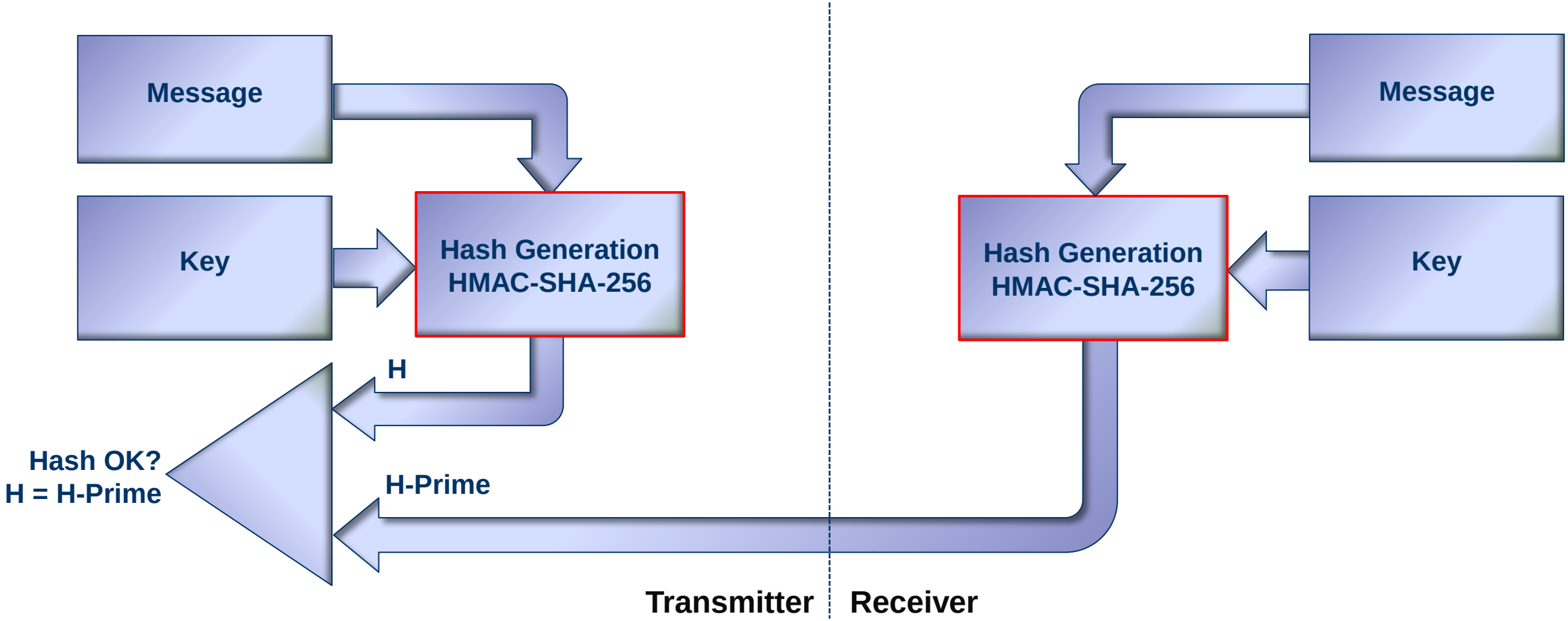
- The ideal cryptographic hash function has **five (5) main properties**:
  - A small change to a message should change the hash value so extensively that the new hash value appears ***uncorrelated*** with the old hash value.
  - It is ***deterministic*** so the same message always results in the same hash digest.
  - It is ***quick to compute*** the hash value for any message.
  - It is ***infeasible*** to generate a message from its hash value ("Digest") except by trying all possible messages.
  - It is infeasible to find two different messages with the same hash value "***collision resistance***."

  Note: "Collision resistance" does not mean that no collisions exist; simply that they are hard to find.

  (Every hash function with more inputs than outputs will necessarily have collisions.)
- In HDCP 2.2 SHA-256 Hash is used for:
  - Verifying the Receiver's Certificate.
  - Verifying that the Receiver properly decrypted the Master Key (Km).
  - Verifying the Locality Check.
  - Storing the Master Key for Pairing.



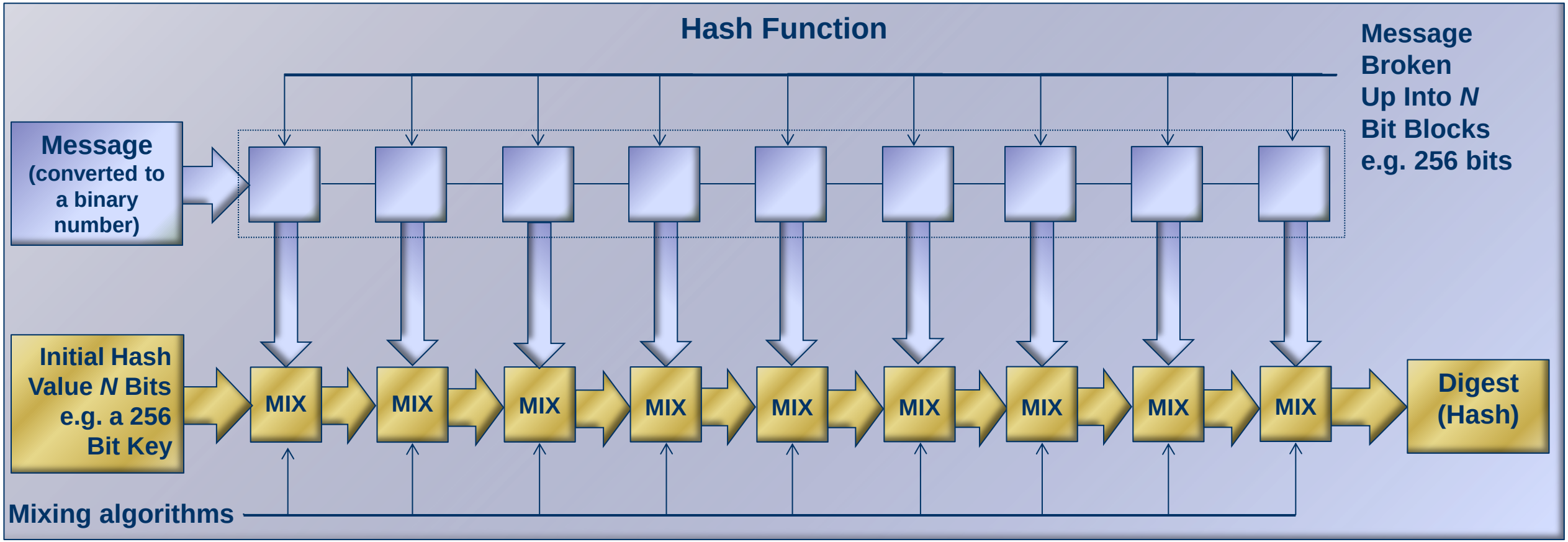
Hash Function Application



Cryptographic Hash Functions – How Do They Work?

Hash Function

- Mixing functions use bitwise operations such as AND, XOR, NOT, OR, Rotate.



Cryptography
Random Number Generation

Cryptographic Random Number Generation

- Types of Random Number Generators:
  - **True** Random Number Generators – Use natural source of randomness such as thermal noise, network statistics, error counter information, etc.
  - **Pseudo**-Random Number Generators – Use initial randomly generated seed value. Uses deterministic algorithms.
- Measure of randomness is “Entropy”:
  - Entropy is the degree of randomness or ***the extent to which all possible outcomes are equally likely***. When entropy is high, it is ***infeasible to predict*** an output better than pure random chance. Dice throws and coin flips offer a high degree of entropy.
  - You can’t infer the entropy from single instance of a “random” bit stream sample. You can only determine the level of entropy from knowing the process itself or empirically through a vast set of trials.

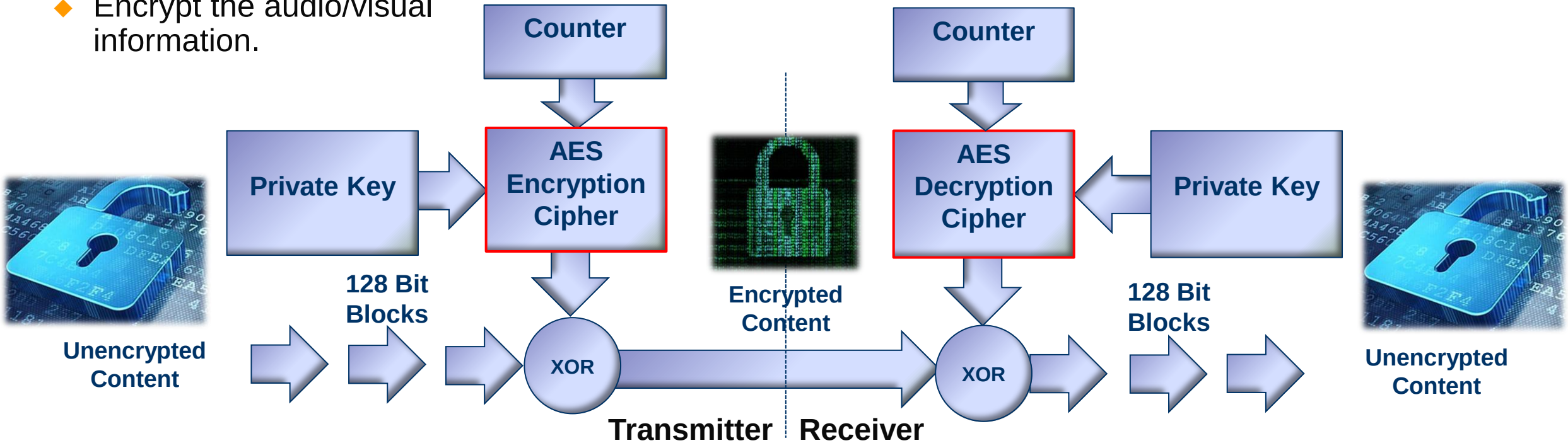
Cryptographic Random Number Generation

- HDCP 2.2 specifies randomness in generating numbers in terms of two levels of Entropy:
  - For generating **R-Tx, R-Rx, Riv, Rn** – Use pseudo-random generation with a *minimum entropy of 40 random bits out of 64-bits*.
  - For generating **Master Key** (Km) and **Session Key** (Ks) – Use ***true*** random number generator or a pseudo random number generator with a ***true random number seed*** (cryptographically secure pseudo random number generator) with *minimum entropy of 128 random bits out of the 128 bits*. This means that the 128 bit Master and Session Keys would have to be true random numbers.
- HDCP recommends NIST SP 800-90 standard for random number generator.

Cryptography
AES Encryption

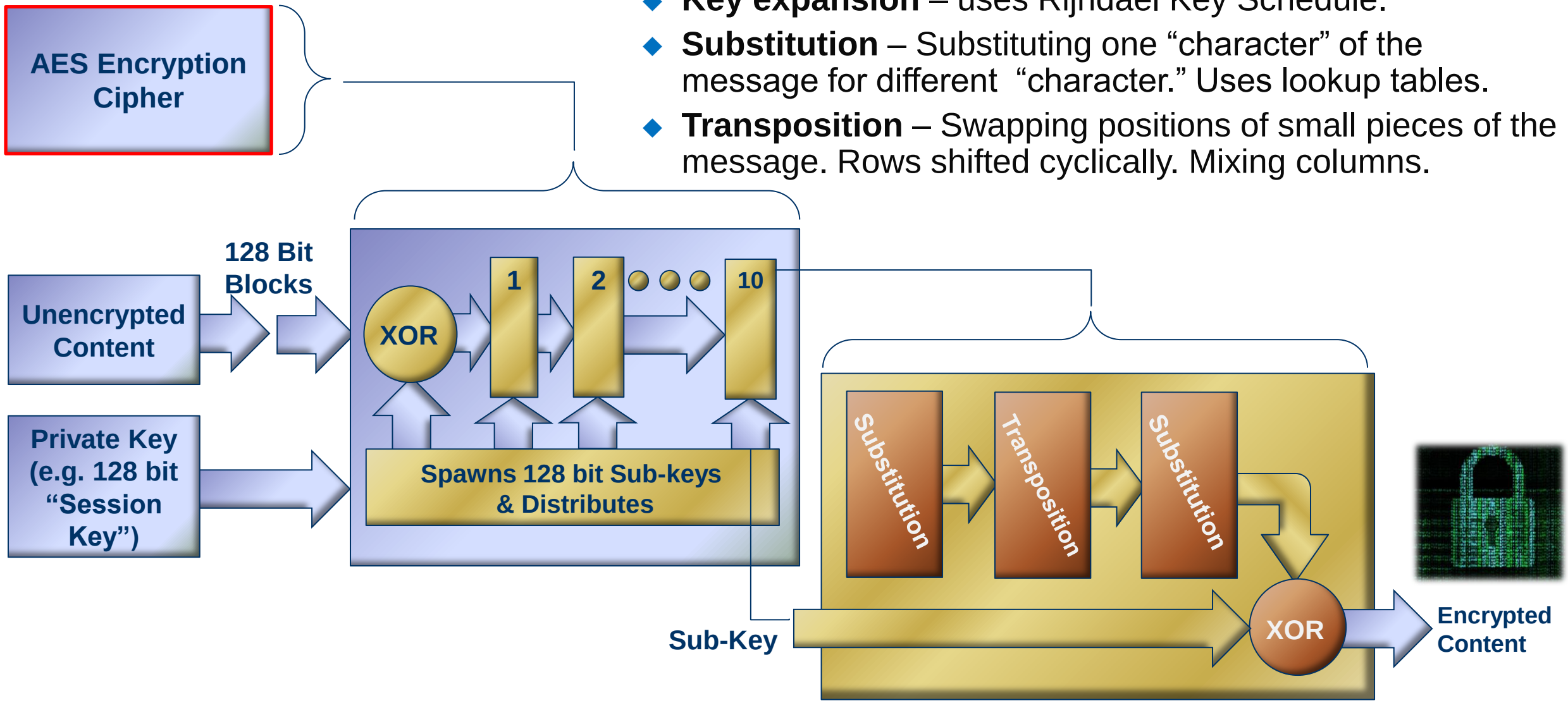
AES Encryption

- AES is a **Symmetric (Private) Key** encryption mechanism. It uses a single shared private key.
- AES is a “block cipher” meaning that it operates on small blocks of data at a time.
- AES has five (5) different modes that it can be operated in. ***HDCP uses Counter Mode*** *(below)*.
- AES is used in HDCP 2.2 for the following functions:
  - Encrypt the Master Key (Km) for storage to facilitate “Pairing.”
  - Encrypting and exchanging the Session Key (Ks).
  - Encrypt the audio/visual information.



AES Encryption – How Does it Work?

- **Key expansion** – uses Rijndael Key Schedule.
- **Substitution** – Substituting one “character” of the message for different “character.” Uses lookup tables.
- **Transposition** – Swapping positions of small pieces of the message. Rows shifted cyclically. Mixing columns.



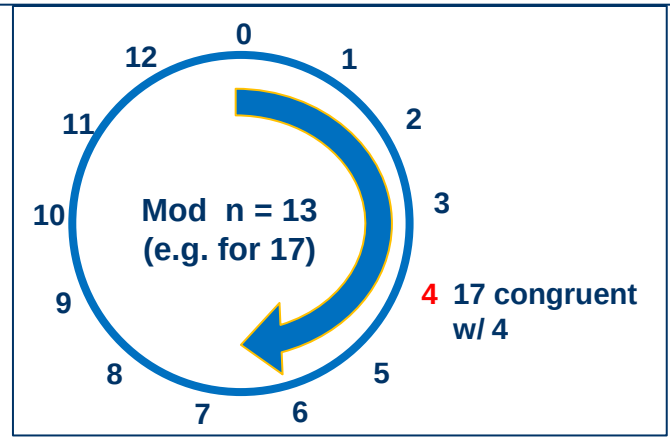
Cryptograpy
RSA Encryption

RSA Encryption

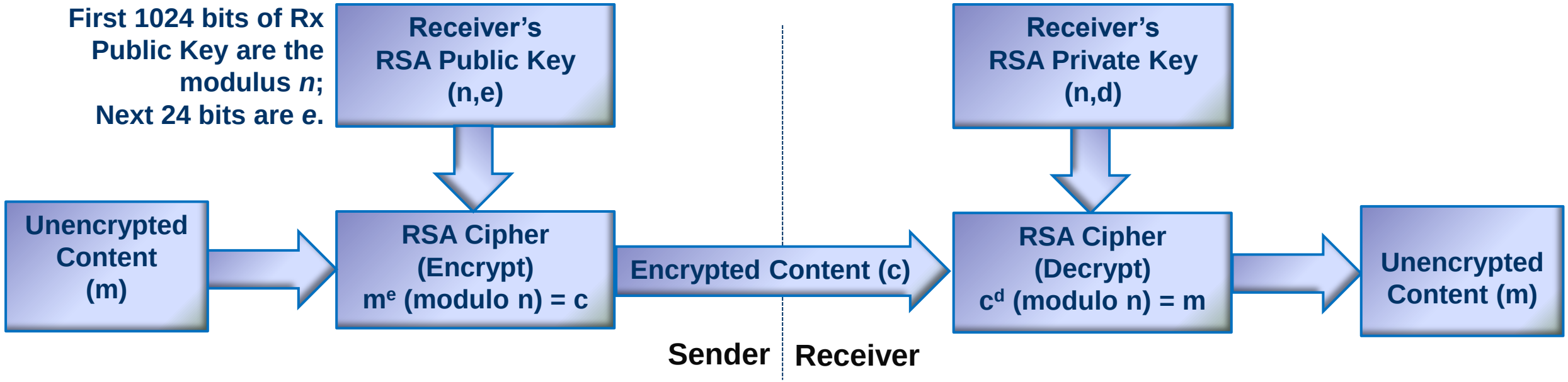
- RSA is the most widely used encryption algorithm.
- RSA is an Asymmetric Key encryption mechanism. It uses a public and a private key.
- RSA is used when there is a need to exchange *small quantities of private information* with multiple parties and *when there is no secure channel* to exchange a private (*shared*) key between parties.
- RSA is used in HDCP 2.2 to *verify the Receiver's certificate* and to *encrypt the Master key* before it is exchanged with a Receiver.
- Developed by MIT professors:

RSA Encryption

- RSA uses the concept of a ***one-way function***; Multiplying 2 prime numbers is easy but *reversing* the process to find the product's 2 prime numbers (prime factorization) is very difficult.
- Uses modular arithmetic, or "clock arithmetic."
- The *n* is the product of 2 large prime numbers; it is used as the modulo in the clock arithmetic.
- Choosing values of e and d is an important factor. Calculation uses what is called a Phi function such that: e * d (mod Phi) = 1.



Modular Arithmetic



Cryptography
Digital Signatures and Digital Certificates

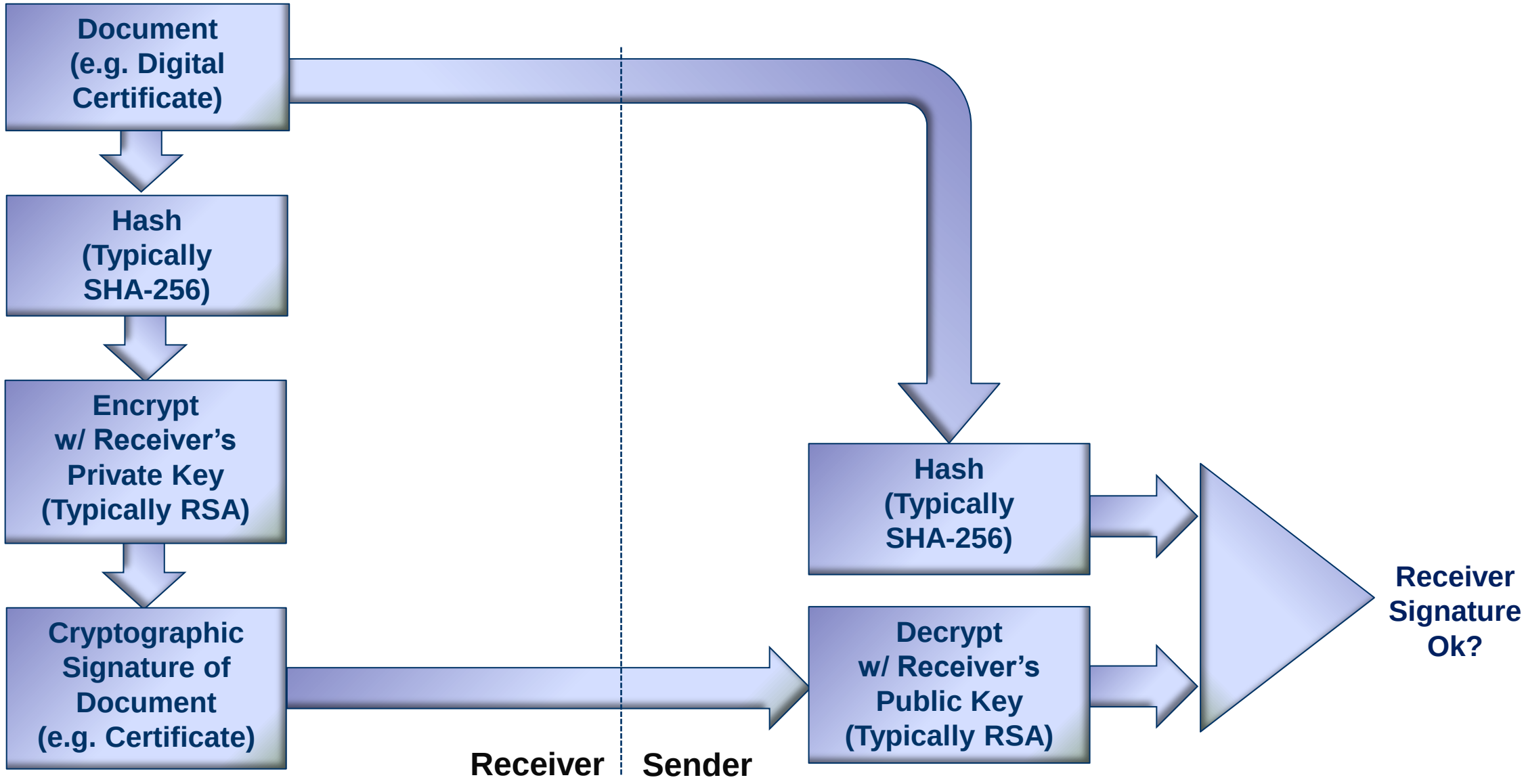
Digital Certificates and Digital Signatures

- Digital Certificates:
  - A Digital Certificate is electronic information or document that is issued by a trusted third party—a “Certificate Authority” (CA)—that provides a way for a sender to verify that a receiver is a trusted party for private communication.
  - Digital Certificates contain an owner identifier and a public key.
- Digital Signatures:
  - A Digital Signature is a *mechanism* or *method* to verify the *authenticity* of a digital message or document (which could be a Digital Certificate).
  - Verifying authenticity means verifying that the message was generated by the assumed sender, i.e. the owner, and that it has not been tampered with.
  - Digital Signatures use RSA (public and private keys) and hashing to affect the signature.
  - It is infeasible to re-generate the Digital Signature without the private key used in the signature.

Digital Certificates and Digital Signatures

- In HDCP 2.2, the Trusted Third party is Digital Content Protection, LLC (DCP) who issues the HDCP Receiver's certificate.
- DCP applies a Digital Signature to the HDCP Receiver's certificate by applying a hash function (Hash-256) and RSA encrypts it with the Receiver's private key.
- This Digital Signature associates the RSA public key (which is part of the certificate that gets hashed and encrypted) with the Receiver ID, i.e. the owner of the Digital Certificate.

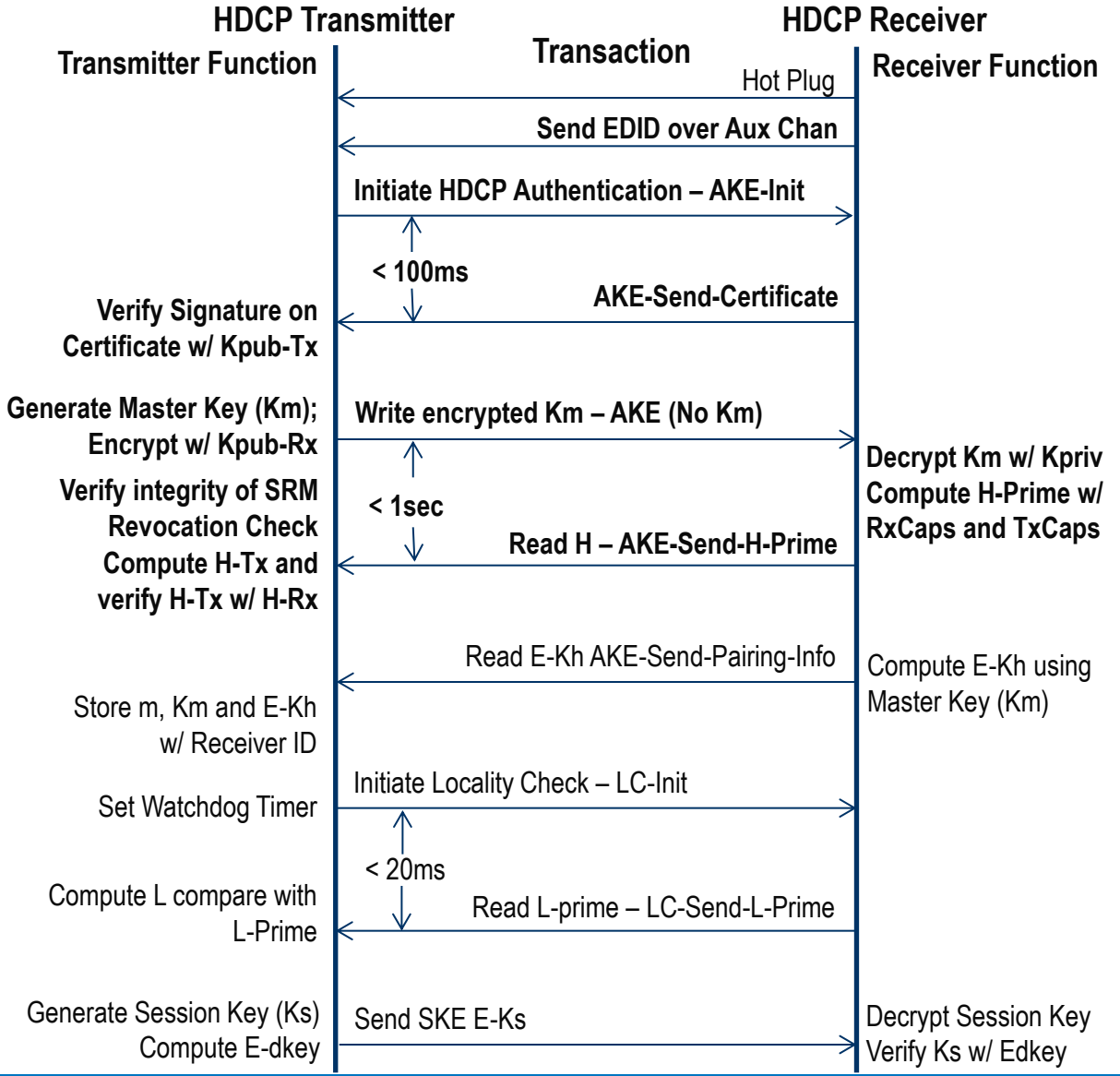
Digital Signatures and Digital Certificates Application



HDCP 2.2 Authentication and Key Exchange

TELEDYNE LECROY
Everywhereyoulook™

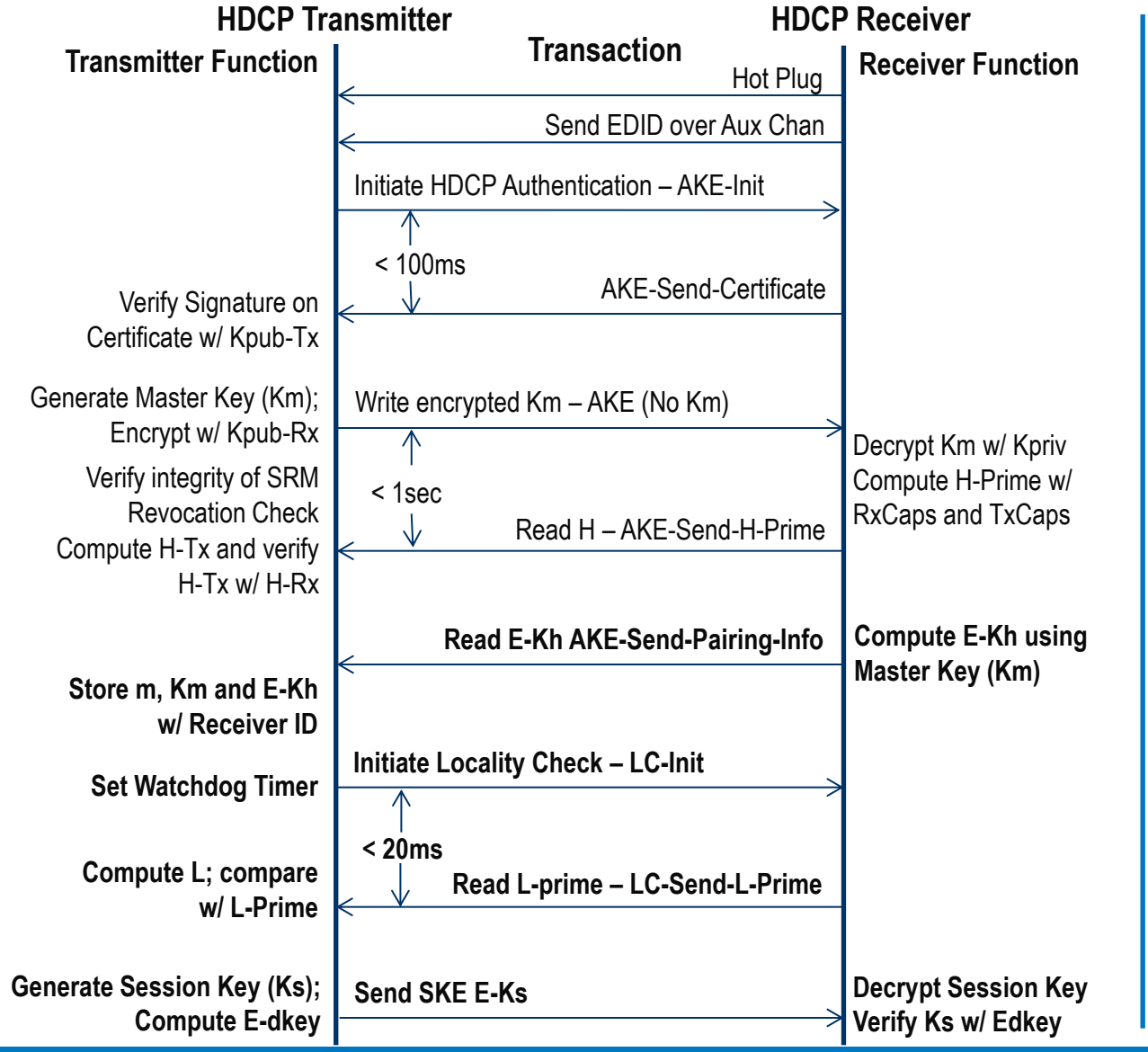
HDCP 2.2 Authentication and Key Exchange Sequence



Authentication and Key Exchange

- Hot plug asserted from the Receiver.
- Transmitter reads EDID from Receiver.
- Initiate Authen. (AKE-Init) – Transmitter sends initiation message (AKE-Init) which contains a 64 bit pseudo-random value (R-Tx) and TxCaps parameters. TxCaps parameters are the HDCP capabilities of the transmitter.
- AKE-Send-Cert – Receiver sends Cert-Rx which contains a 64 bit pseudo-random value (R-Rx) and RxCaps parameters in response to Transmitter read. RxCaps parameters are the HDCP capabilities of the receiver. (Must be transmitted within 100 ms of AKE-Init.)
- Transmitter extracts Receiver ID from Cert-Rx - Verifies Signature on Cert-Rx using Kpub-Rx (a 1048 bit RSA public key of a receiver) using the following steps:
  - Transmitter generates Km (a 128-bit Master Key). Km is then encrypted using E-Kpub (a 1024 bit value).
  - Write encrypted Km-AKE – Transmitter sends AKE-Km message to receiver containing the E-Kpub.
  - Receiver decrypts Km w/ Kpriv-Rx (Receiver private key RSA).
  - Receiver computes H-prime (256-bit) w/ RxCaps & TxCaps.
  - Read H (AKE-Send-H-prime) – Receiver sends AKE-Send-H-Prime in response to Transmitter read. H-prime must be within 1 Sec from time the Transmitter writes AKE-Km message.
  - Transmitter verifies receiver with revocation list.
- Transmitter computes H-Tx; compares with H-Rx (H-Prime).

HDCP 2.2 Authentication…Pairing, Locality Check, Session Key Exchange



Pairing

- Transmitter-Receiver Pairing is performed using the following steps:
  - Read E-Kh – AKE-Send-Pairing-Info – Receiver sends E-Kh using encryption of Km in response to Transmitter read.
  - Transmitter stores m, Km and E-Kh with Receiver ID (a 40 bit value [20 ones and 20 zeros) that uniquely identifies a licensed receiver).

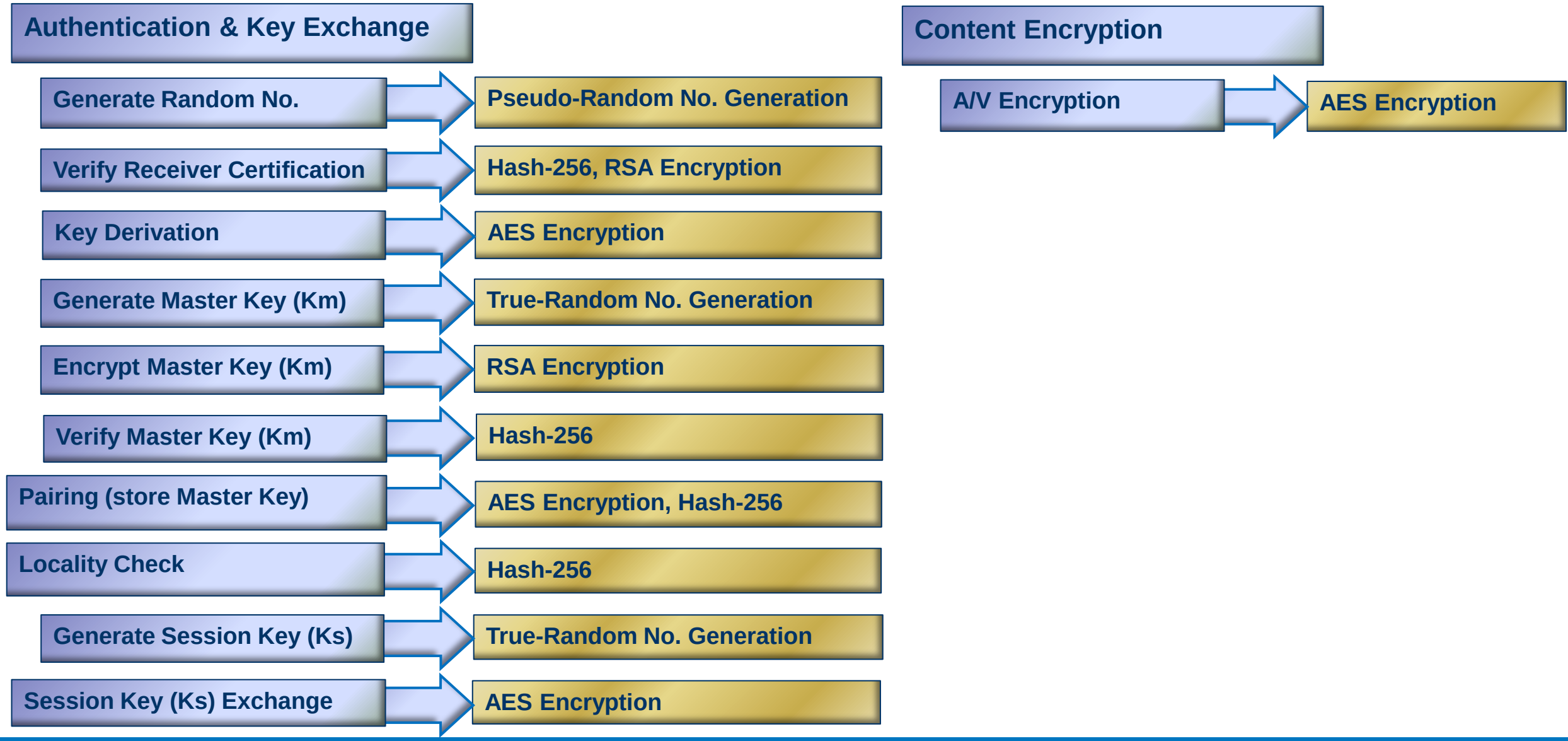
Locality Check

- Locality Check is performed by Transmitter using the following steps:
  - Transmitter sets a watchdog timer.
  - Initiate Locality Check – Transmitter sends LC-Init to receiver.
  - Receiver computes L-Prime (256 bit value).
  - Read L - Receiver transmits LC-Send-L-Prime in response to Transmitter read.
  - Transmitter computes L and compares w/ L-prime from receiver.

Session Key Exchange

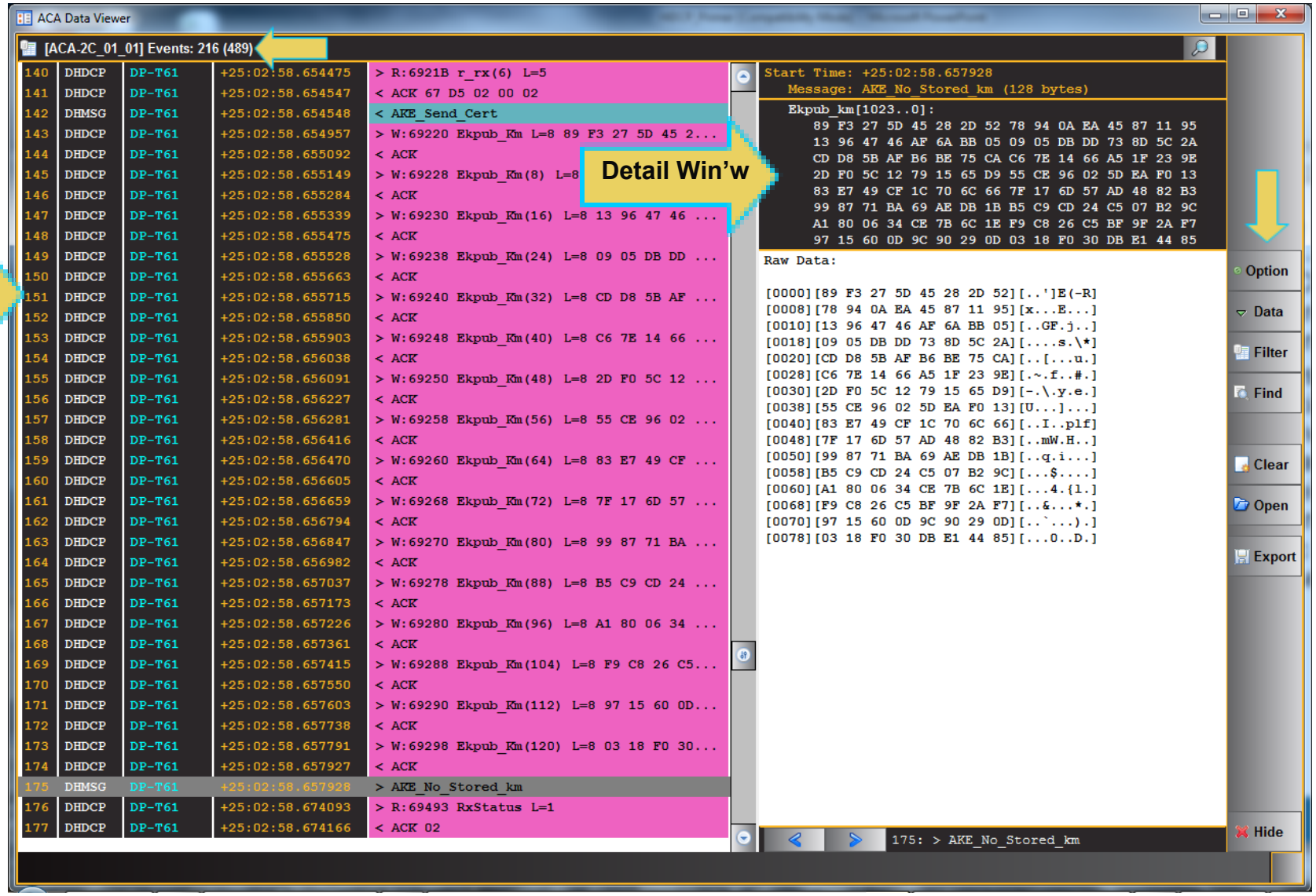
- Session Key Exchange involves the following steps:
  - Transmitter generates a 128-bit pseudo-random Session key (Ks) and a 64-bit pseudo-random number R-iv.
  - Derives/computes 128-bit E-dkey using Ks.
  - Transmitter sends E-dkey to receiver.
  - Receiver derives Ks, verifies that it's equal to E-dkey received.

HDCP 2.2 Cryptographic Summary



Brief Introduction to Aux Channel Protocol Analzyer

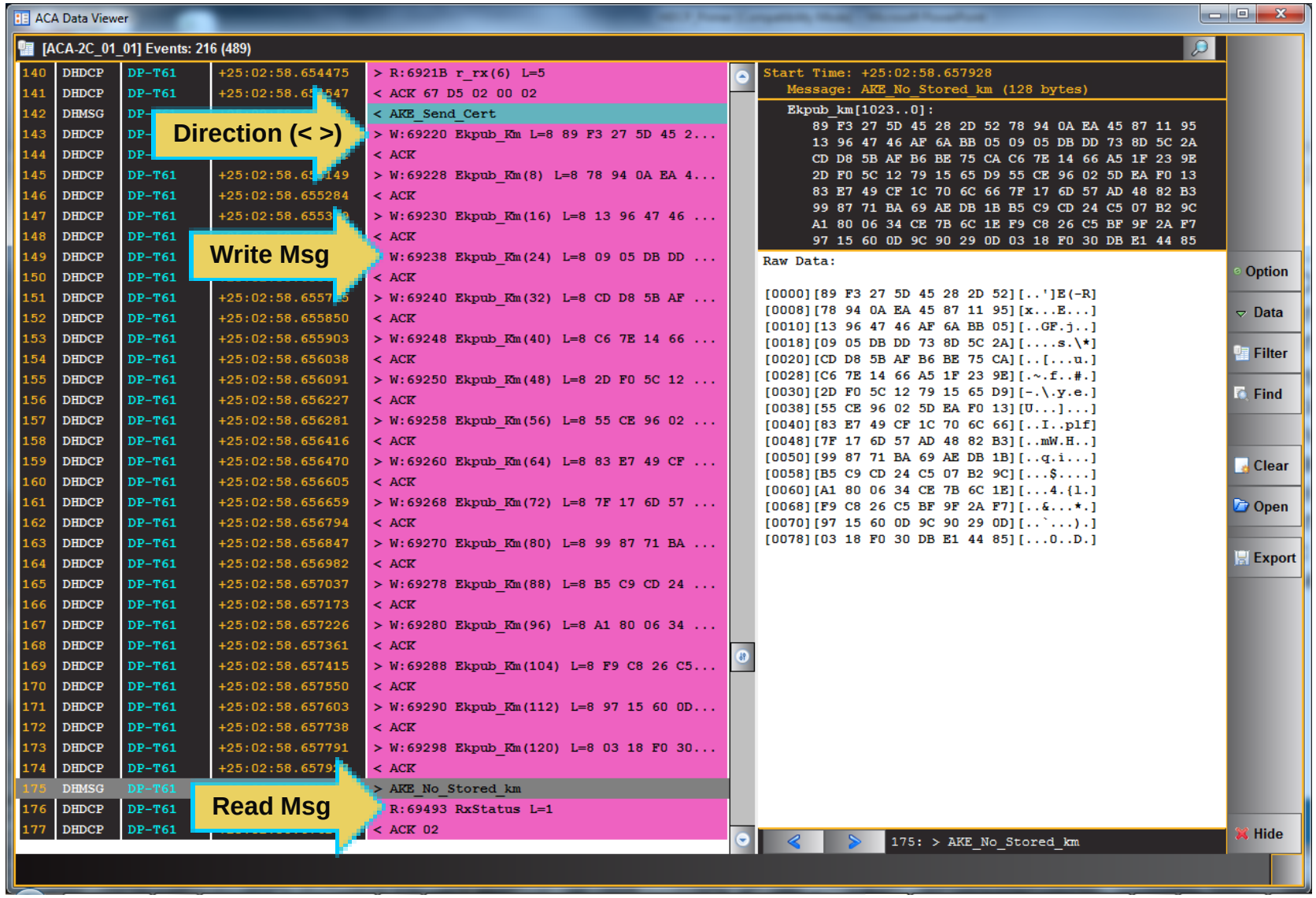
Auxiliary Channel Analyzer (ACA) Utility



140	DHDCP	DP-T61	+25:02:58.654475	> R:6921B r_rx(6) L=5
141	DHDCP	DP-T61	+25:02:58.654547	< ACK 67 D5 02 00 02
142	DHMSG	DP-T61	+25:02:58.654548	< AKE_Send_Cert
143	DHDCP	DP-T61	+25:02:58.654957	> W:69220 Ekpub_Km L=8 89 F3 27 5D 45 2...
144	DHDCP	DP-T61	+25:02:58.655092	< ACK
145	DHDCP	DP-T61	+25:02:58.655149	> W:69228 Ekpub_Km(8) L=8
146	DHDCP	DP-T61	+25:02:58.655284	< ACK
147	DHDCP	DP-T61	+25:02:58.655339	> W:69230 Ekpub_Km(16) L=8 13 96 47 46 ...
148	DHDCP	DP-T61	+25:02:58.655475	< ACK
149	DHDCP	DP-T61	+25:02:58.655528	> W:69238 Ekpub_Km(24) L=8 09 05 DB DD ...
150	DHDCP	DP-T61	+25:02:58.655663	< ACK
151	DHDCP	DP-T61	+25:02:58.655715	> W:69240 Ekpub_Km(32) L=8 CD D8 5B AF ...
152	DHDCP	DP-T61	+25:02:58.655850	< ACK
153	DHDCP	DP-T61	+25:02:58.655903	> W:69248 Ekpub_Km(40) L=8 C6 7E 14 66 ...
154	DHDCP	DP-T61	+25:02:58.656038	< ACK
155	DHDCP	DP-T61	+25:02:58.656091	> W:69250 Ekpub_Km(48) L=8 2D F0 5C 12 ...
156	DHDCP	DP-T61	+25:02:58.656227	< ACK
157	DHDCP	DP-T61	+25:02:58.656281	> W:69258 Ekpub_Km(56) L=8 55 CE 96 02 ...
158	DHDCP	DP-T61	+25:02:58.656416	< ACK
159	DHDCP	DP-T61	+25:02:58.656470	> W:69260 Ekpub_Km(64) L=8 83 E7 49 CF ...
160	DHDCP	DP-T61	+25:02:58.656605	< ACK
161	DHDCP	DP-T61	+25:02:58.656659	> W:69268 Ekpub_Km(72) L=8 7F 17 6D 57 ...
162	DHDCP	DP-T61	+25:02:58.656794	< ACK
163	DHDCP	DP-T61	+25:02:58.656847	> W:69270 Ekpub_Km(80) L=8 99 87 71 BA ...
164	DHDCP	DP-T61	+25:02:58.656982	< ACK
165	DHDCP	DP-T61	+25:02:58.657037	> W:69278 Ekpub_Km(88) L=8 B5 C9 CD 24 ...
166	DHDCP	DP-T61	+25:02:58.657173	< ACK
167	DHDCP	DP-T61	+25:02:58.657226	> W:69280 Ekpub_Km(96) L=8 A1 80 06 34 ...
168	DHDCP	DP-T61	+25:02:58.657361	< ACK
169	DHDCP	DP-T61	+25:02:58.657415	> W:69288 Ekpub_Km(104) L=8 F9 C8 26 C5...
170	DHDCP	DP-T61	+25:02:58.657550	< ACK
171	DHDCP	DP-T61	+25:02:58.657603	> W:69290 Ekpub_Km(112) L=8 97 15 60 0D...
172	DHDCP	DP-T61	+25:02:58.657738	< ACK
173	DHDCP	DP-T61	+25:02:58.657791	> W:69298 Ekpub_Km(120) L=8 03 18 F0 30...
174	DHDCP	DP-T61	+25:02:58.657927	< ACK
175	DHMSG	DP-T61	+25:02:58.657928	> AKE_No_Stored_km
176	DHDCP	DP-T61	+25:02:58.674093	> R:69493 RxStatus L=1
177	DHDCP	DP-T61	+25:02:58.674166	< ACK 02

- The Name of transaction log file on top banner.
- Control buttons are on the right.
- Two main panels:
  - Transaction Log Panel
  - Detail Panel (shows details for selected transaction)

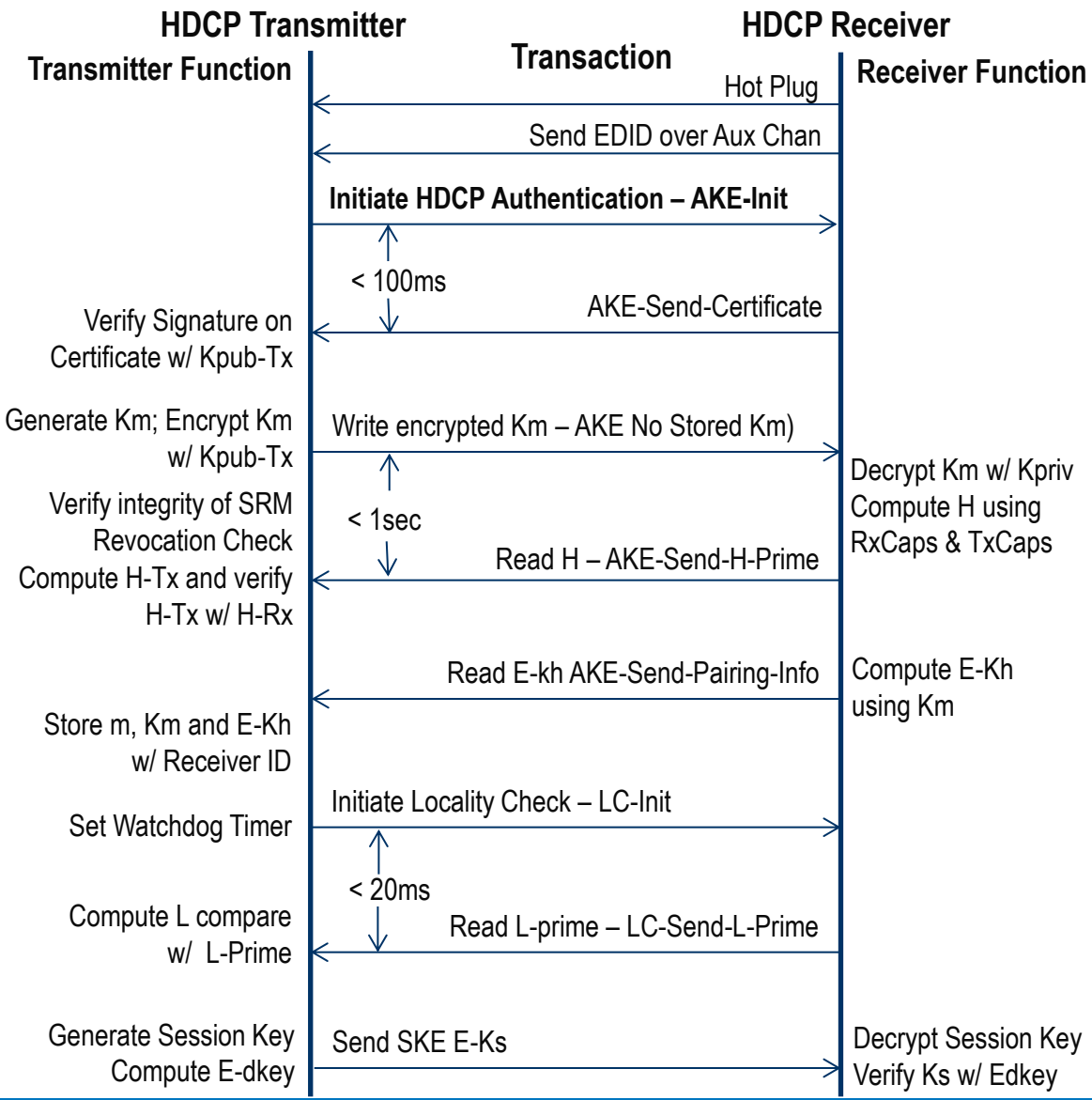
Auxiliary Channel Analyzer (ACA) Utility

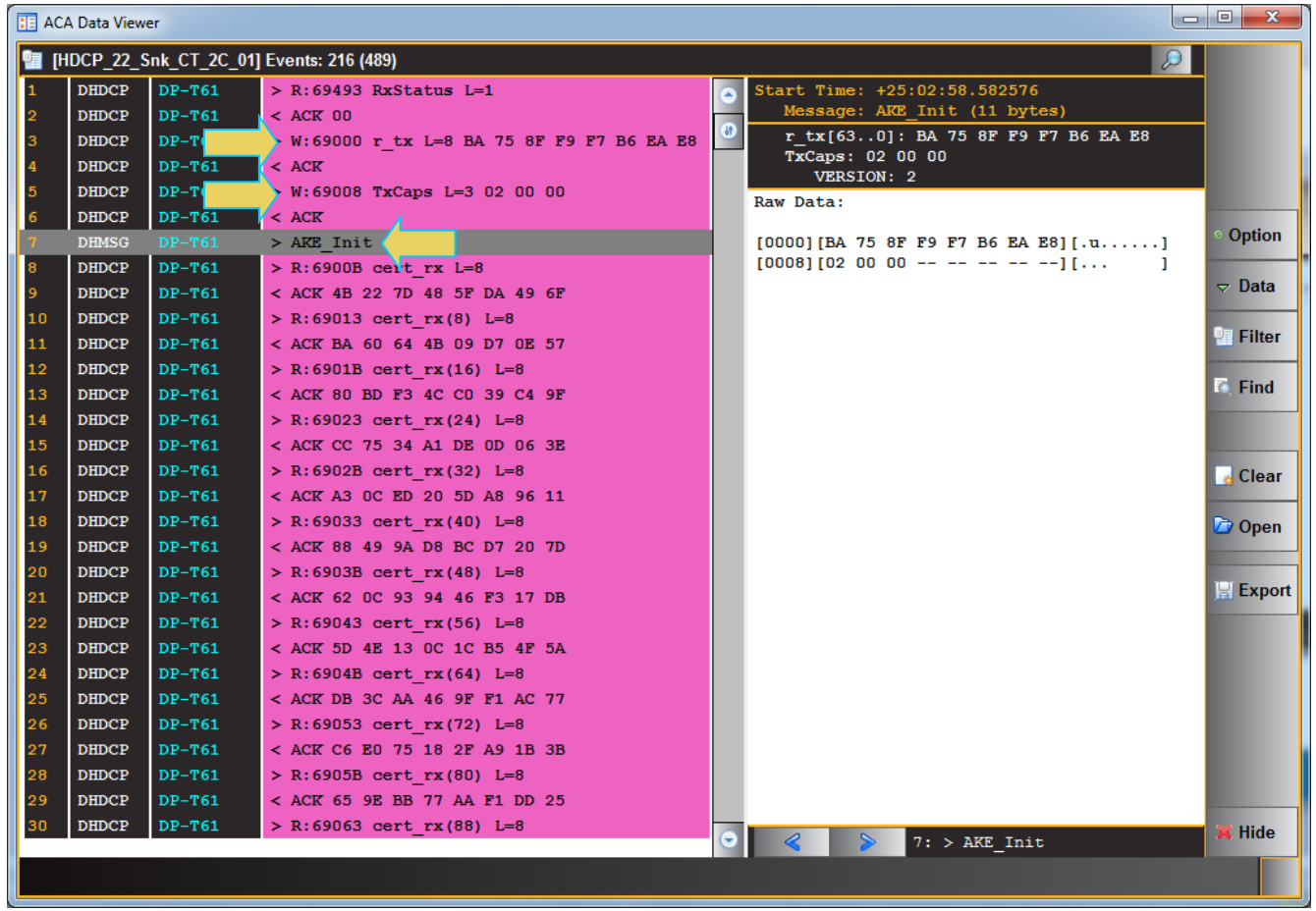


- HDCP transactions can be logged from either a 980 module's Tx port(s) or the module's Rx port.
- HDCP transactions can either be Reads ("R:") or Writes ("W:") or an acknowledgement ("ACK").
- The "<" or ">" indicate the direction of transmission from the perspective of the monitoring port. In this case the monitoring port is the 980 Tx. The > means a message going from the 980 Tx to the sink DUT.

HDCP Authentication
Verifying the Receiver Certificate

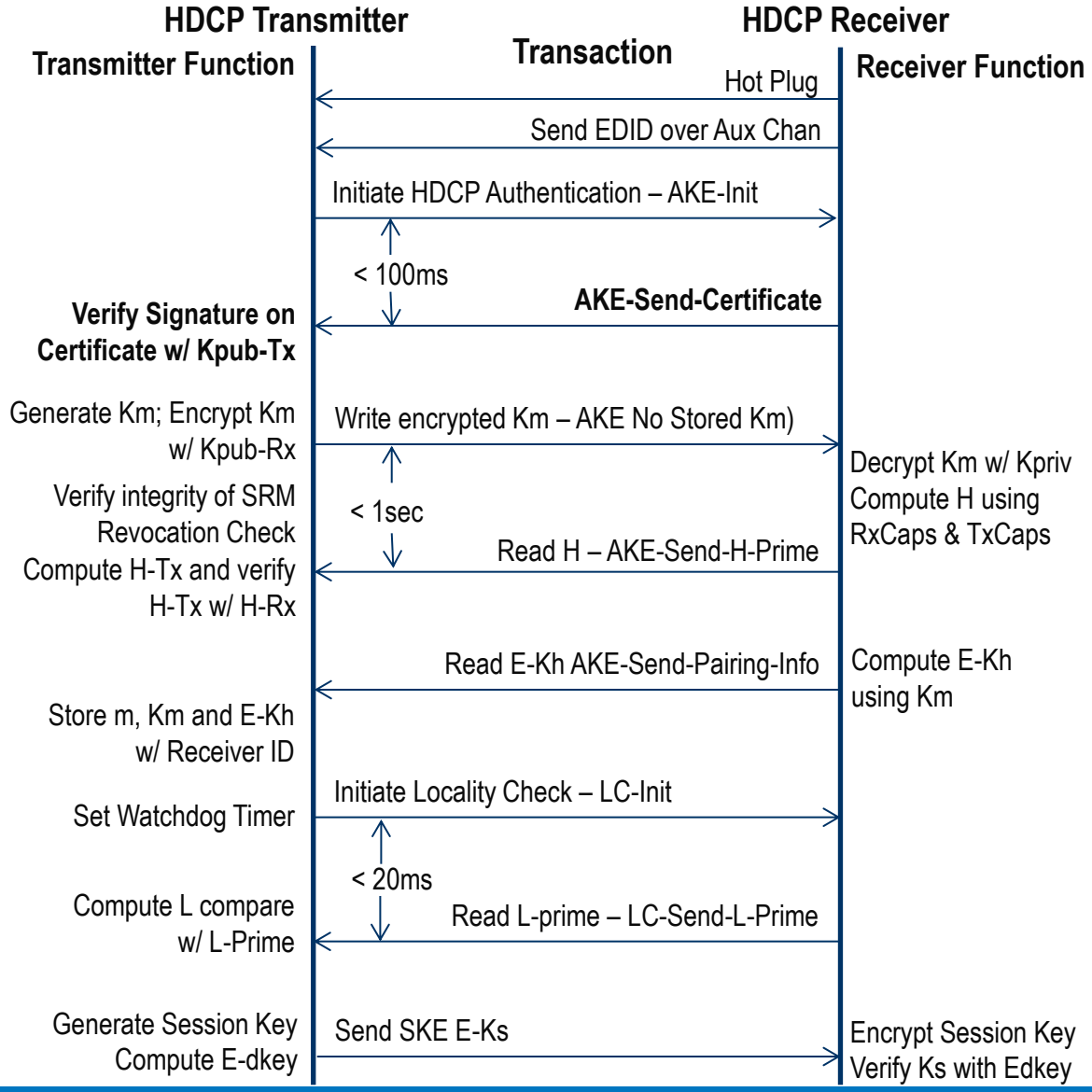
HDCP 2.2 Sequence – AKE Initiation, Send Random Number and TxCaps

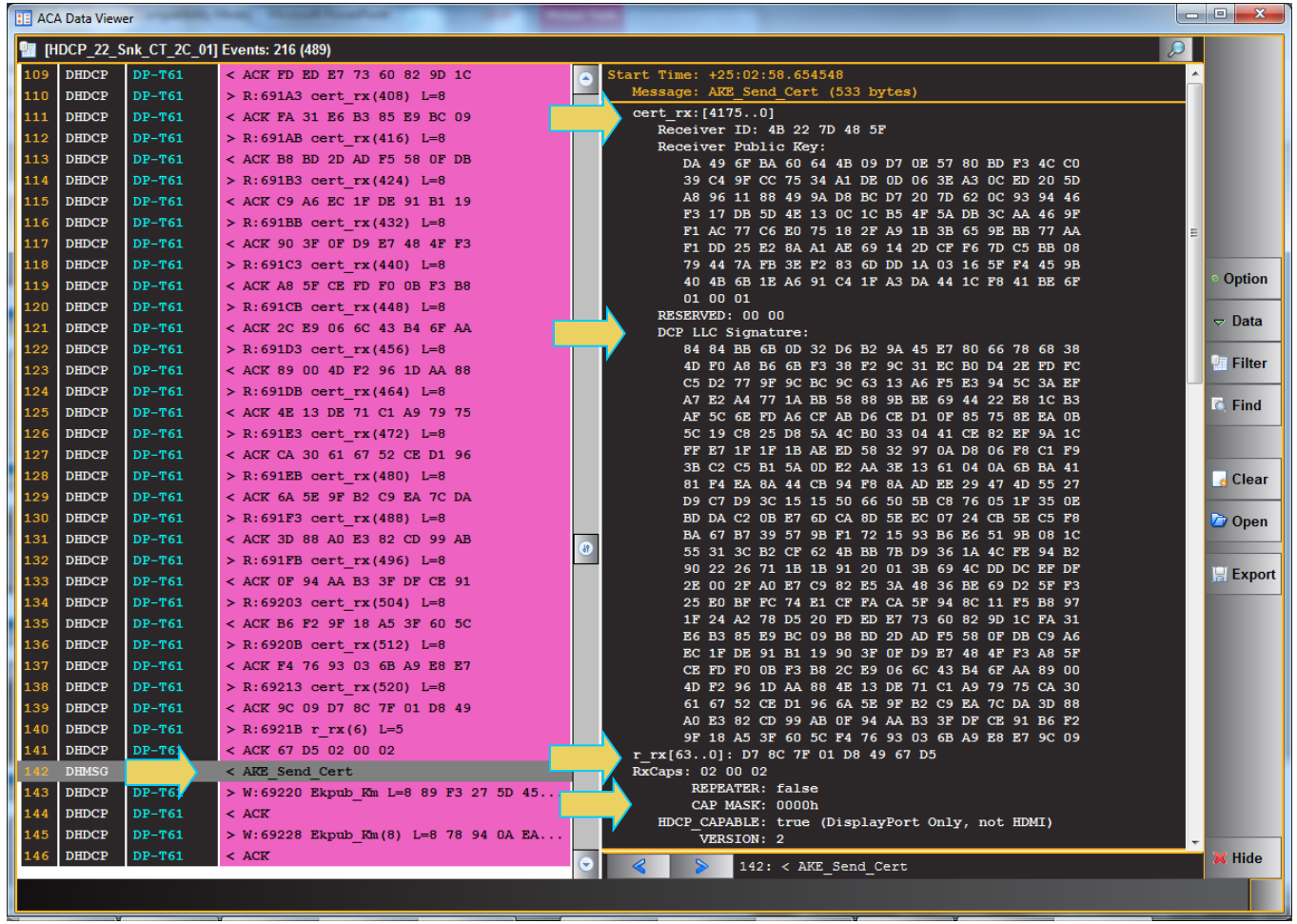




- Transmitter initiates Authentication and Key Exchange
  - Sends R-Tx (64 bit pseudo-random number) later used in the encryption of the Master Key (Km).
  - Sends TCaps (Transmitter HDCP capabilities [Version]).

HDCP 2.2 Sequence – Transmitter Reads Receiver Certificate

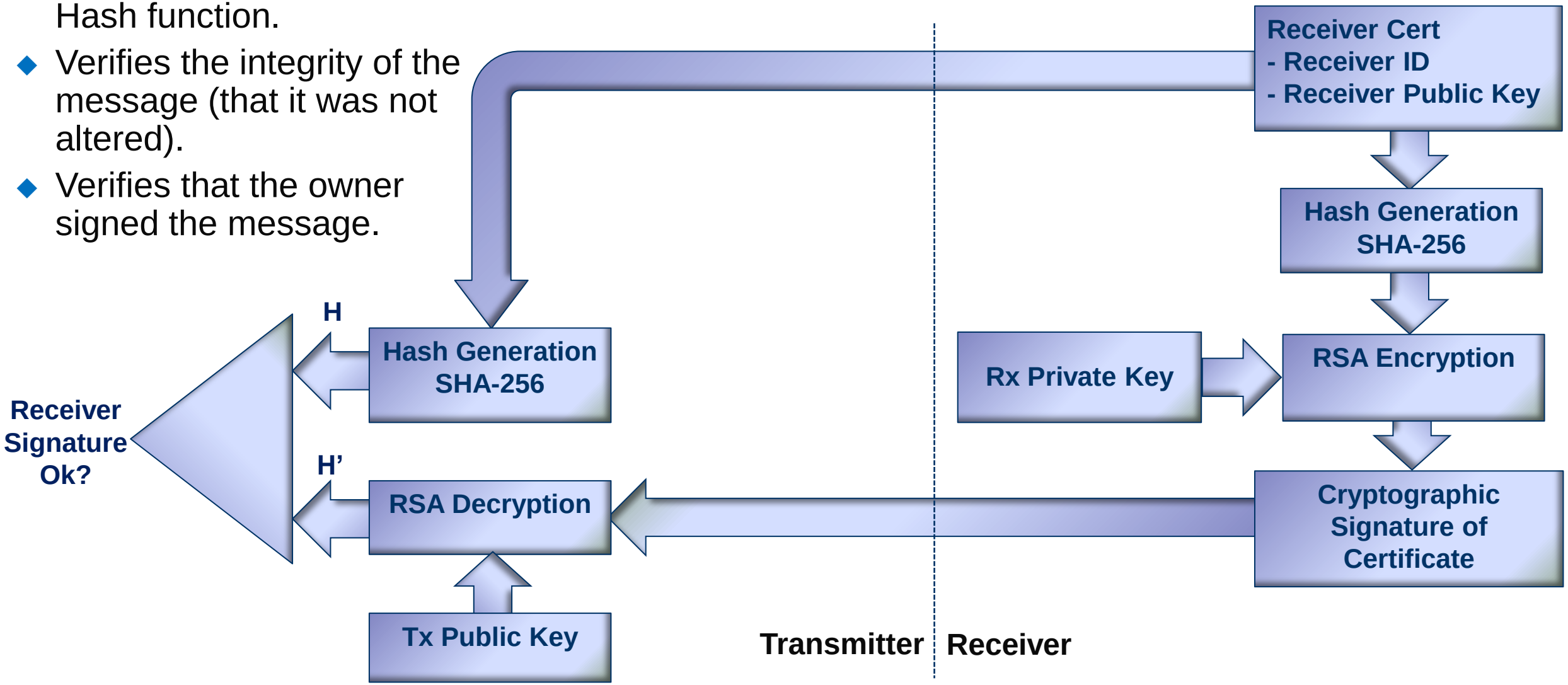




- Transmitter reads Receiver Certificate. Certificate contains:
  - Receiver ID – Unique receiver ID; 40 bits: 20 ones & 20 zeros
  - Receiver Public Key – Unique 1040 bit RSA public key ($kpub_{rx}$).
  - DCP signature – Calculated over all fields in certificate.

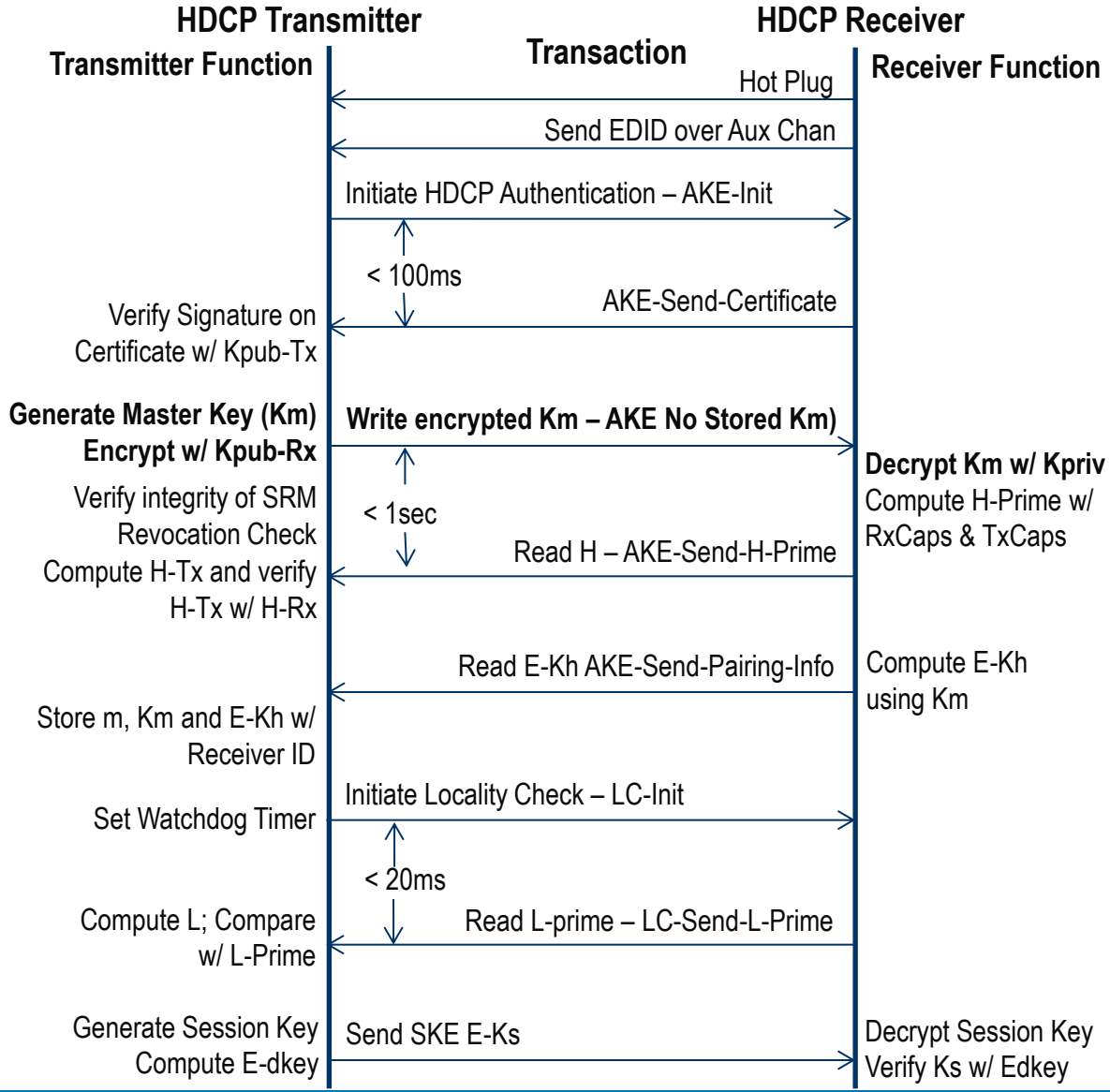
HDCP 2.2 – Verifying Signature on Receiver Certificate

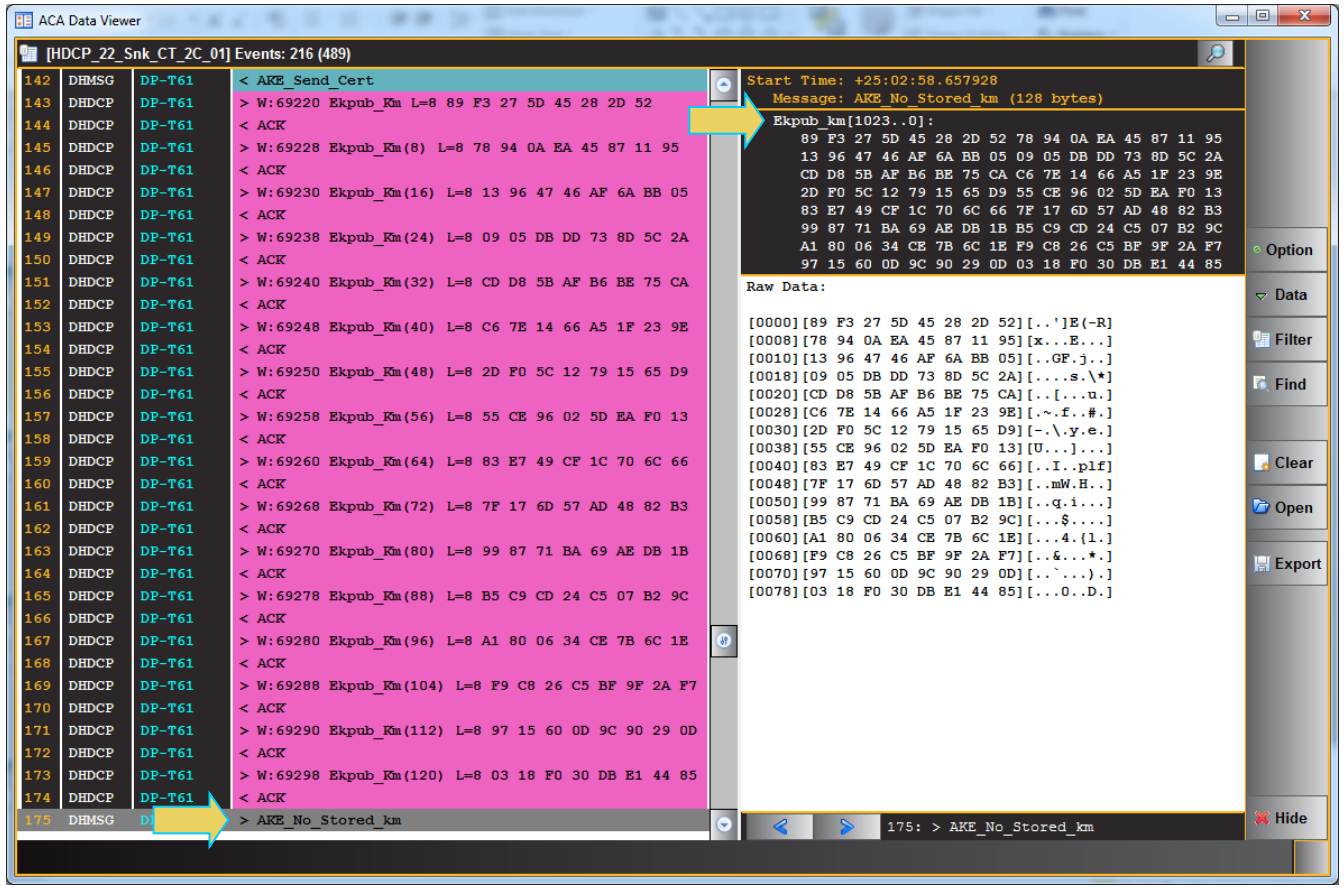
- Transmitter verifies that the Receiver's certificate is authentic using Hash function.
- Verifies the integrity of the message (that it was not altered).
- Verifies that the owner signed the message.



HDCP Authentication
Generating, Exchanging the Master Key

HDCP 2.2 Sequence – Transmitter Sends Encrypted Master Key to Receiver

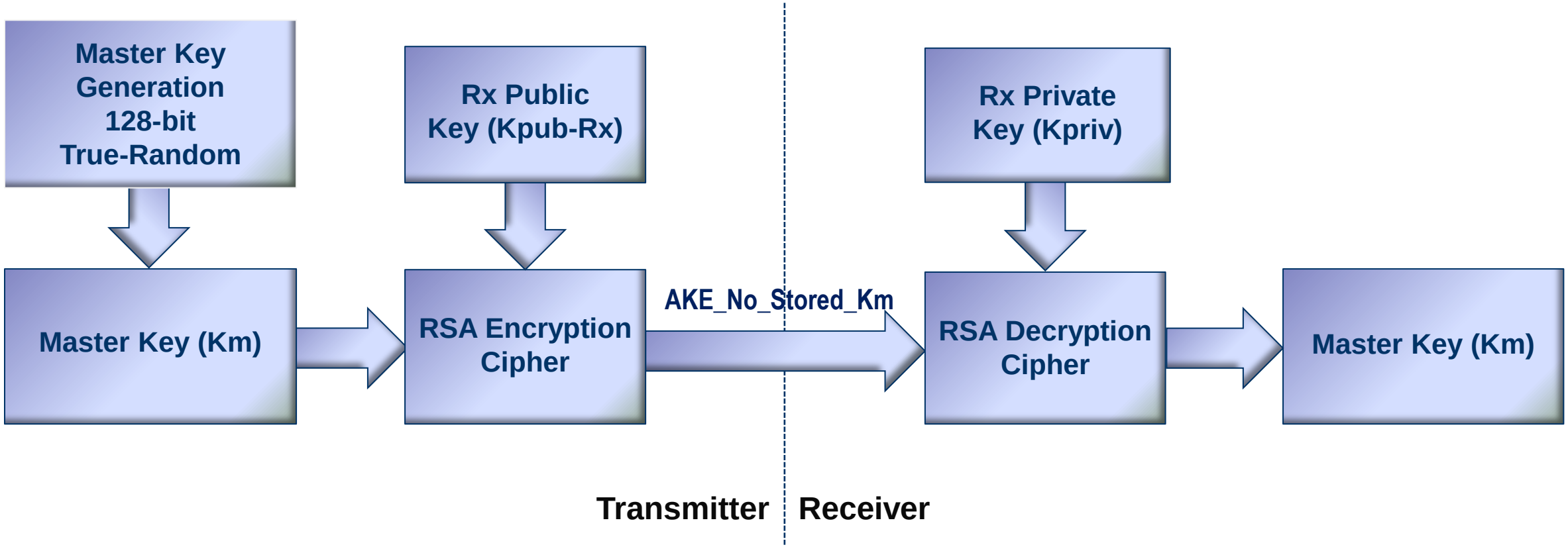




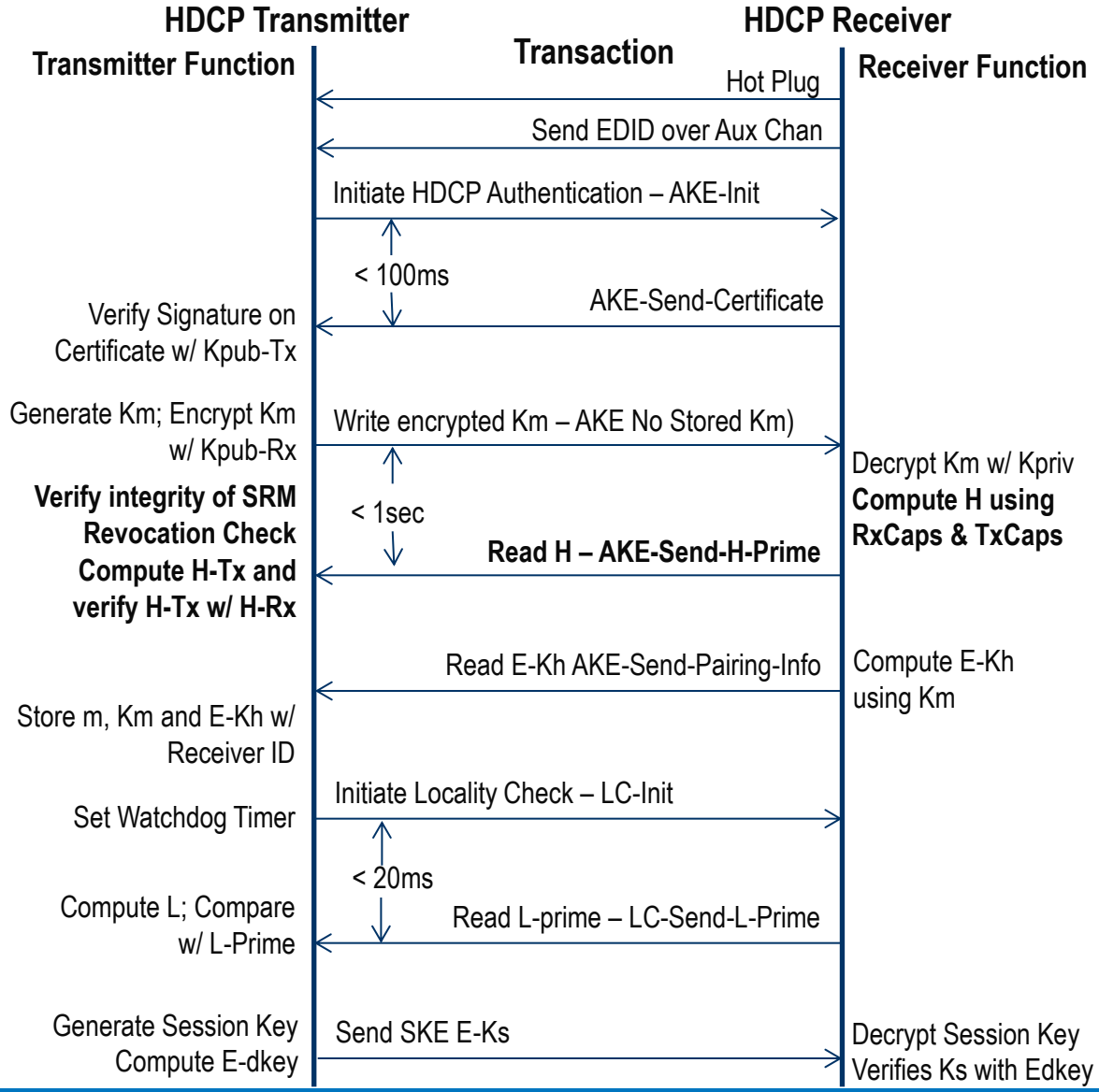
- Transmitter writes Ekpub_Km message with Master Key to Receiver (since there is no stored Master Key [Km]):

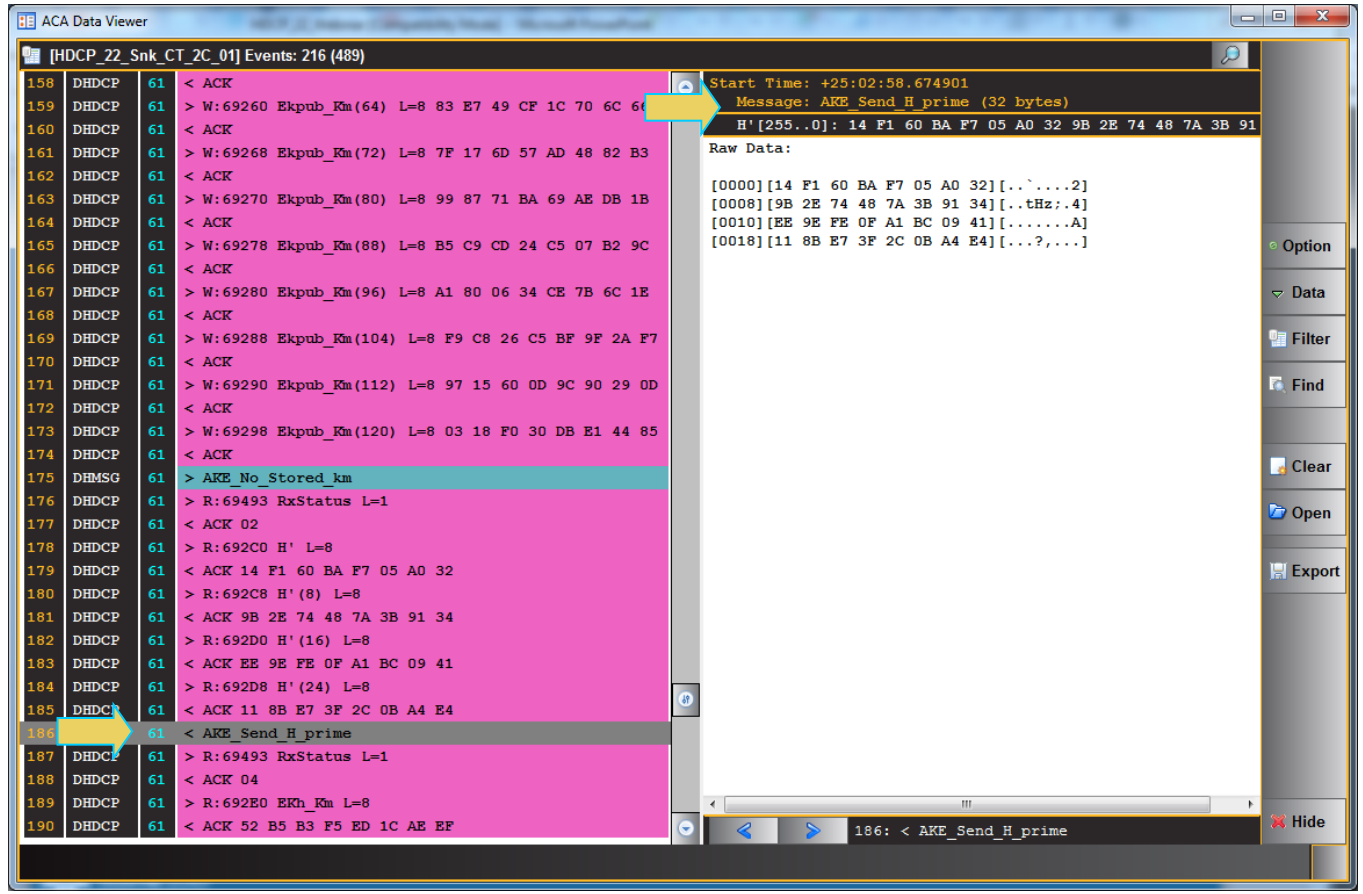
HDCP 2.2 – Generating, Encrypting and Exchanging Master Key

- Master Key (Km) is a 128-bit key produced by the Transmitter's True-random number generator.
- Master Key is encrypted with the Receiver's public key and sent to the Receiver.



HDCP 2.2 Sequence – Transmitter Reads Receiver H-Prime

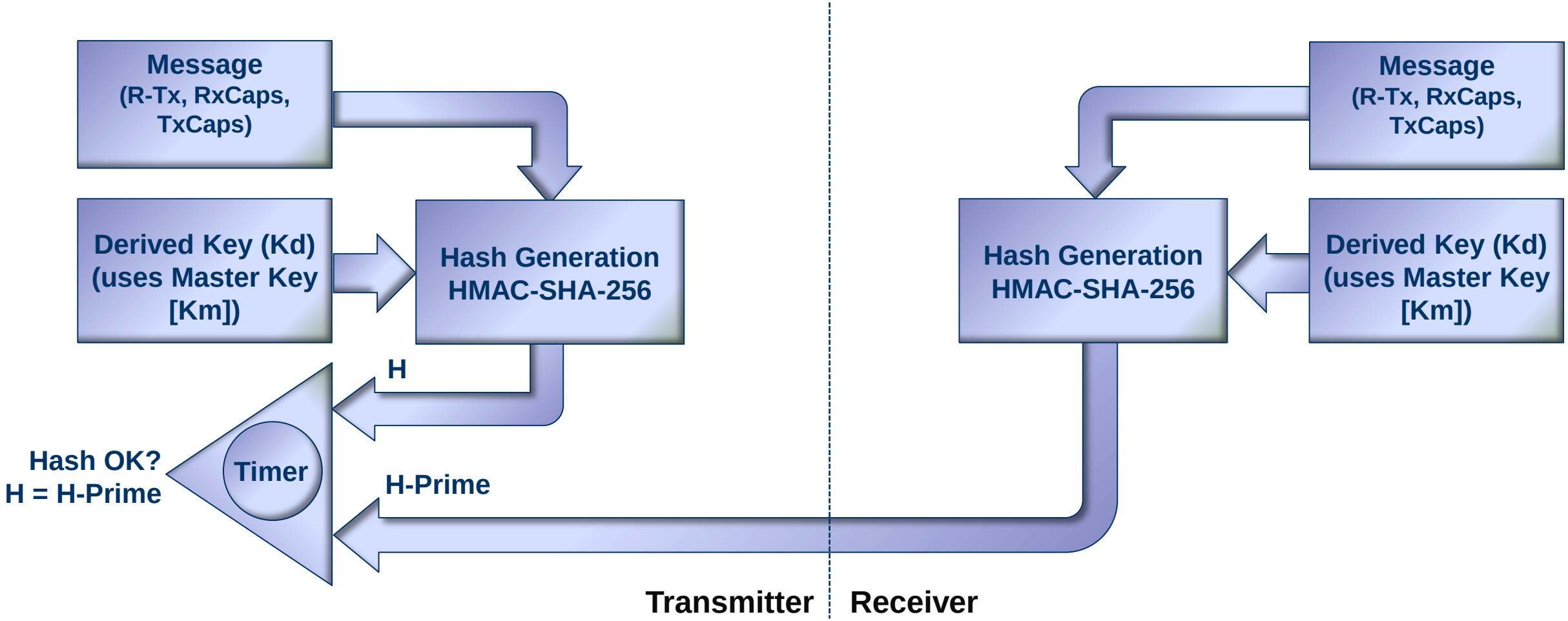




- Transmitter reads Receiver's H-Prime.
- This verifies the integrity of the Master Key, i.e. that it was received and decrypted properly.

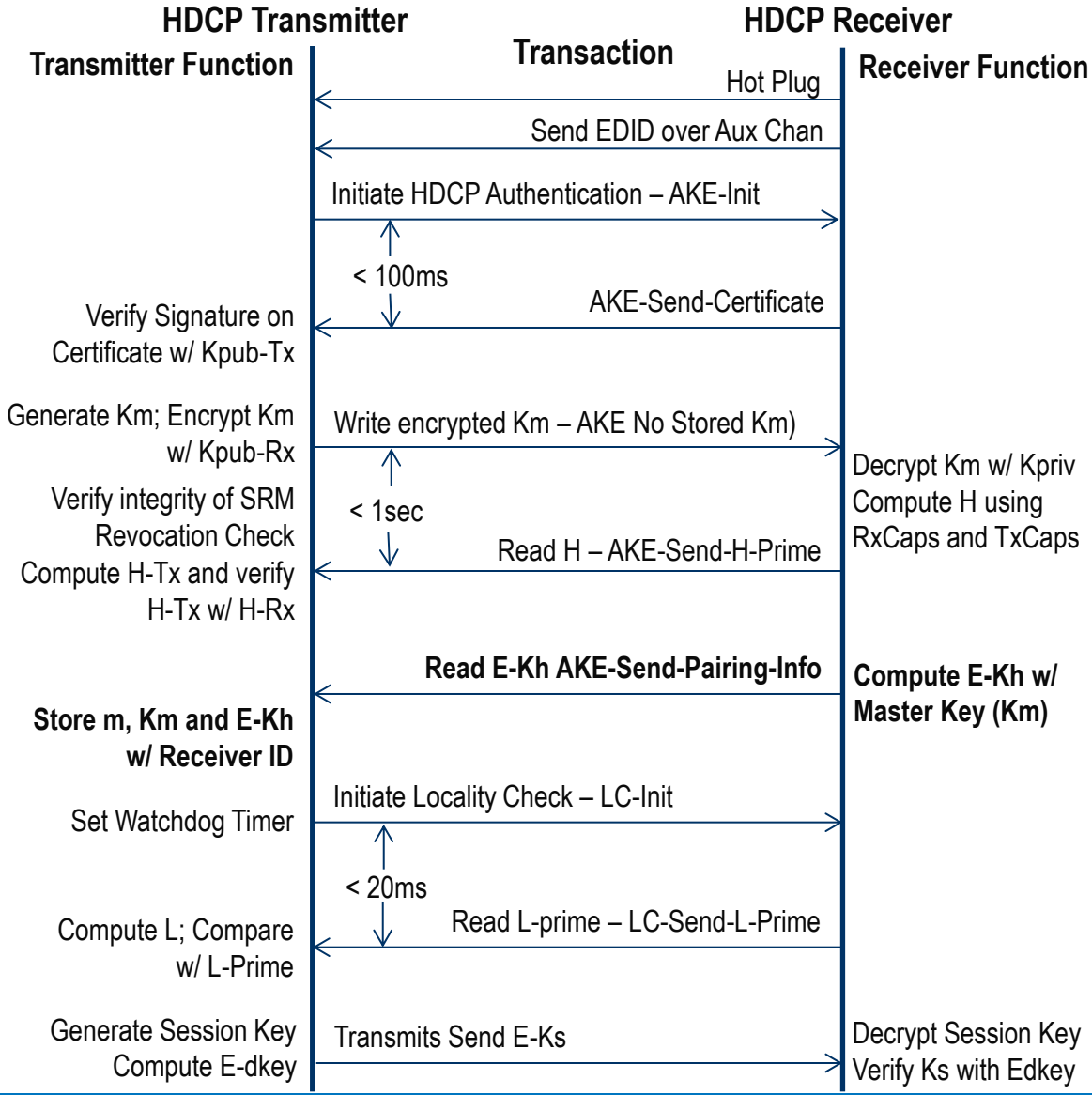
HDCP 2.2 – Verifying Master Key Exchange & Integrity with H=H-Prime

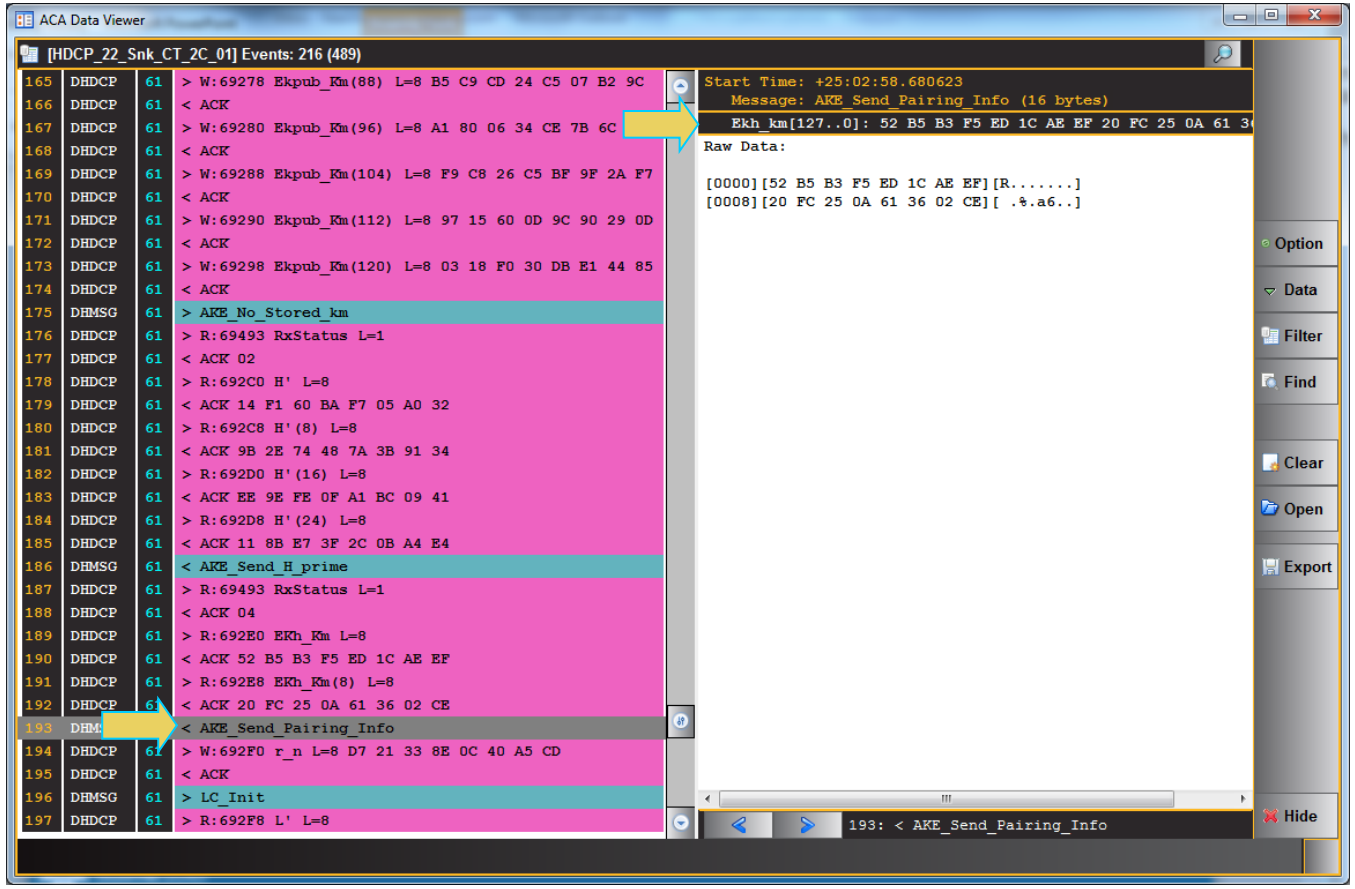
- Master Key is verified by using it on both the Transmitter and Receiver in a hash function to hash a message comprised of RTx, RxCaps and TxCaps.



HDCP Authentication Pairing

HDCP 2.2 Sequence – Transmitter Reads Pairing Info

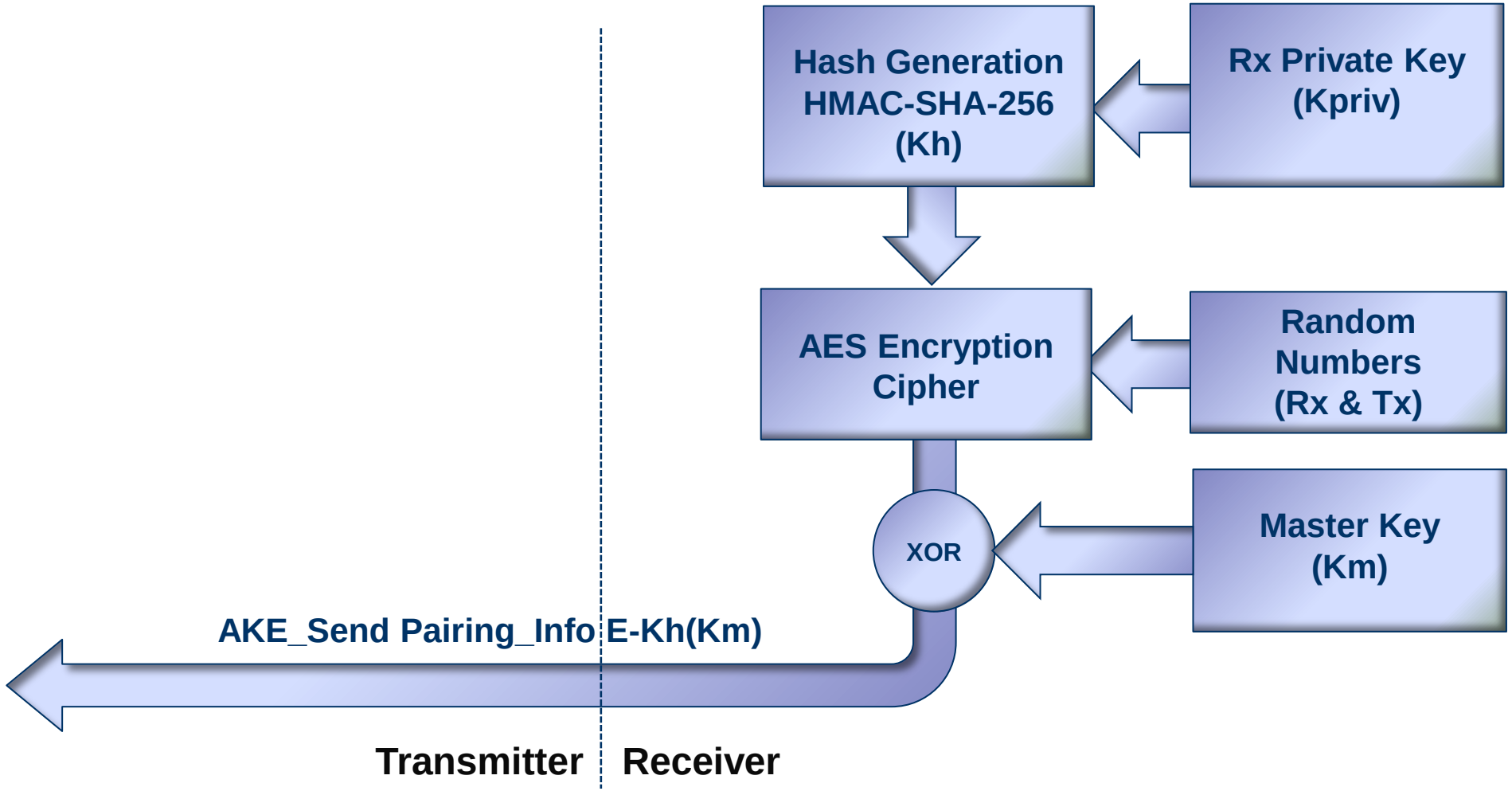




- Transmitter reads Receiver's E-Kh AKE Send Pairing info.

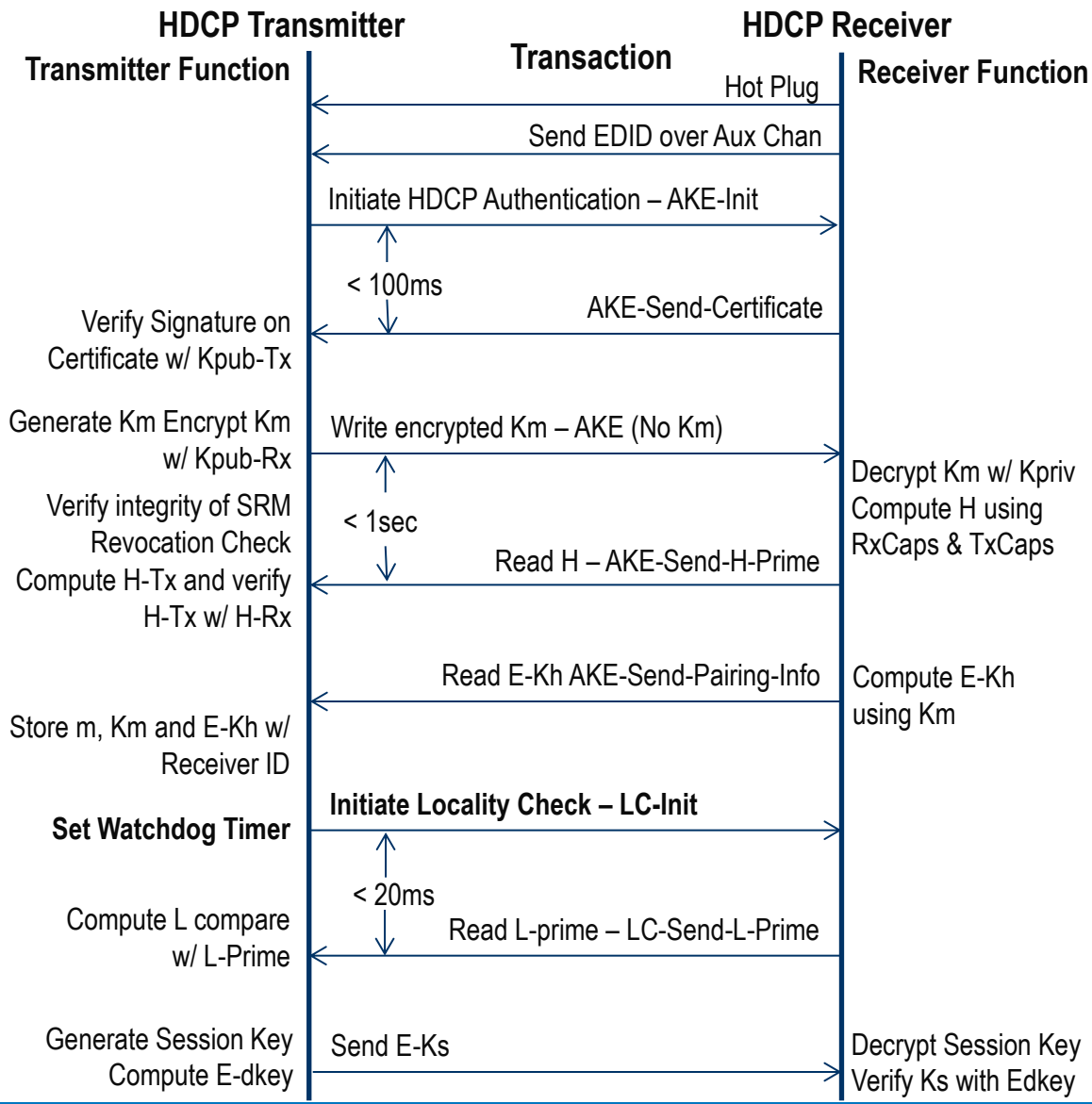
HDCP 2.2 – Master Key Storage for Pairing

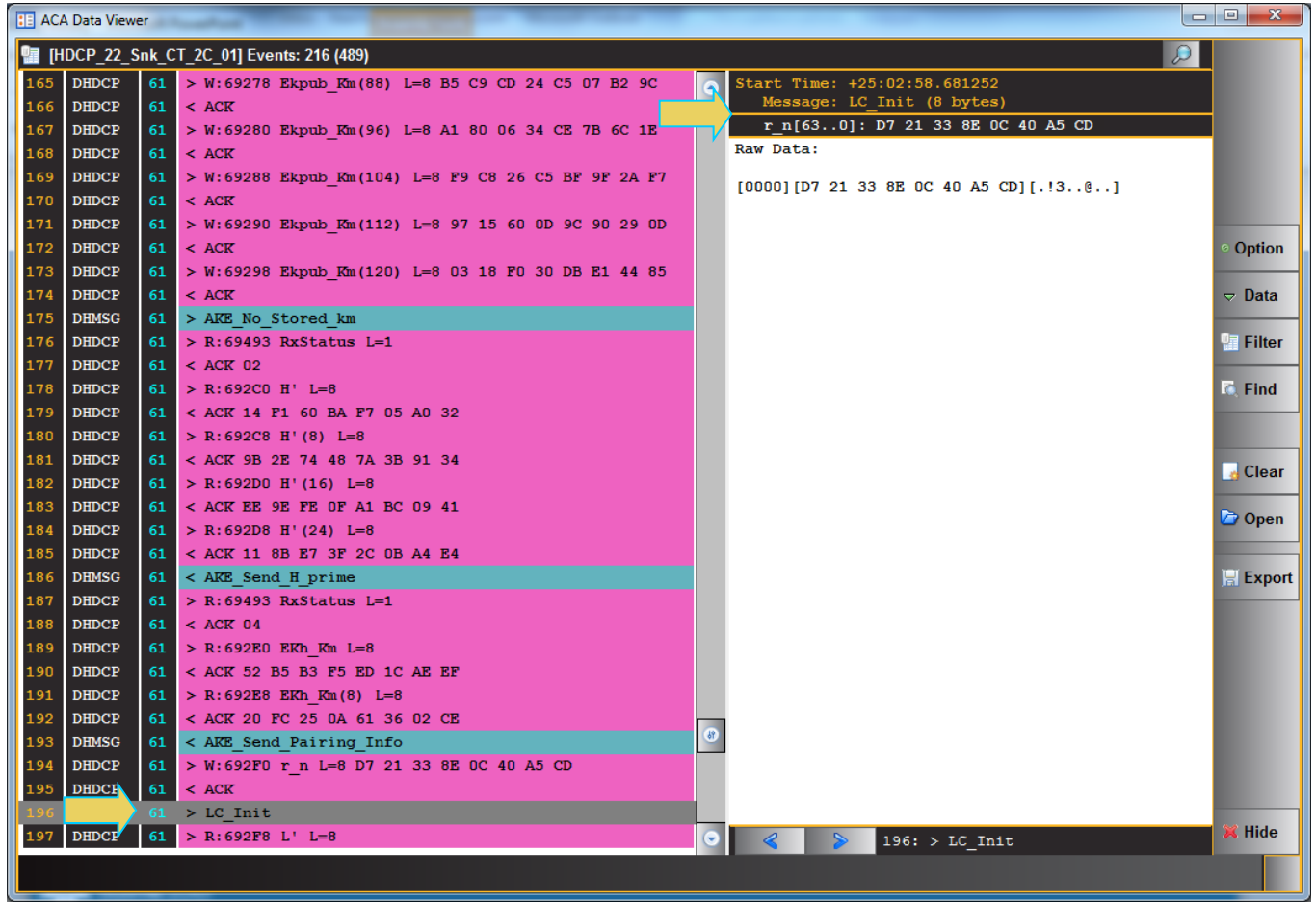
- Master Key (Km) is encrypted by Receiver and sent to Transmitter for storing to support pairing the next time these two devices authenticate together.



HDCP Authentication Locality Check

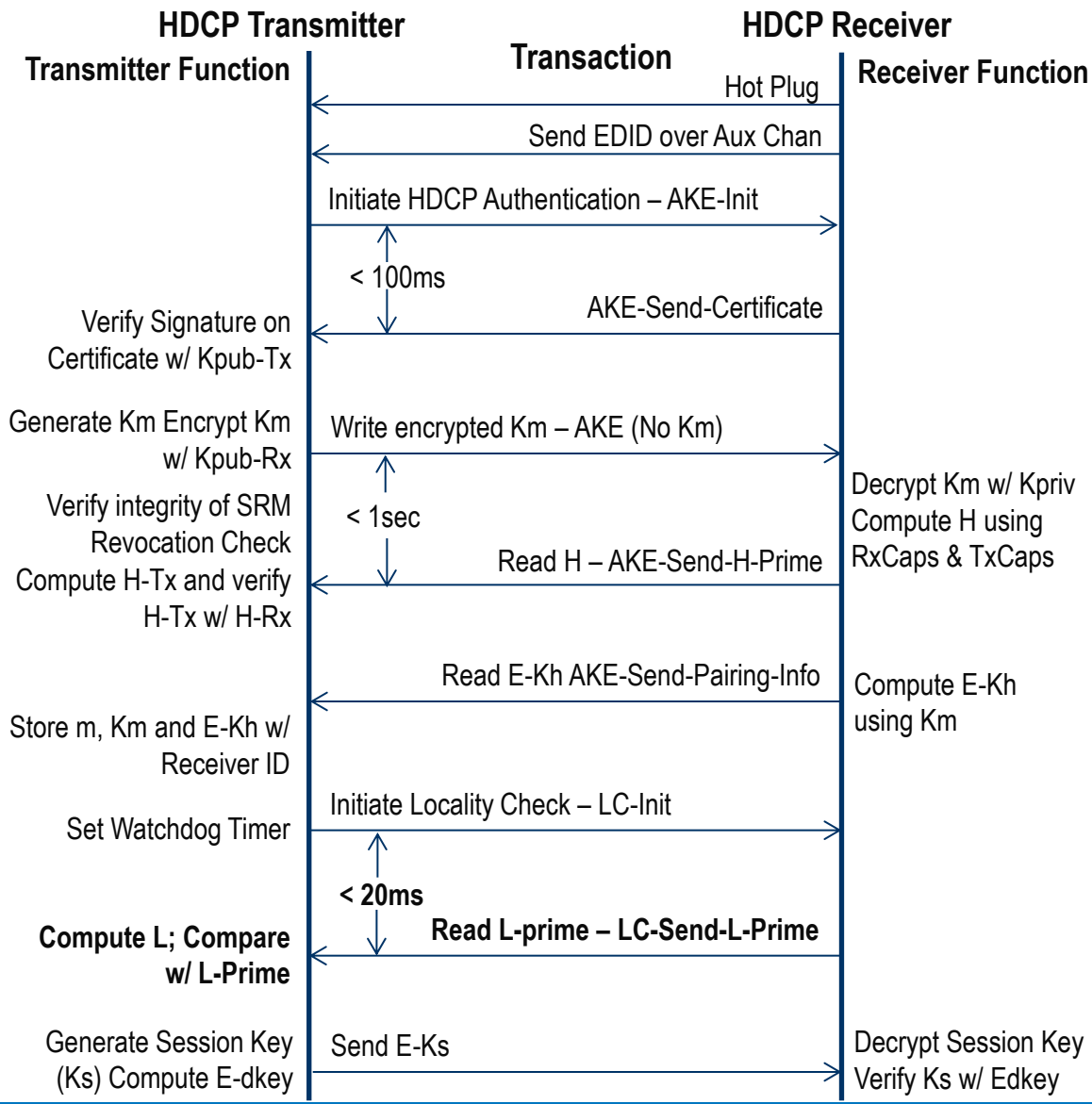
HDCP 2.2 Sequence – Transmitter Initiates Locality Check

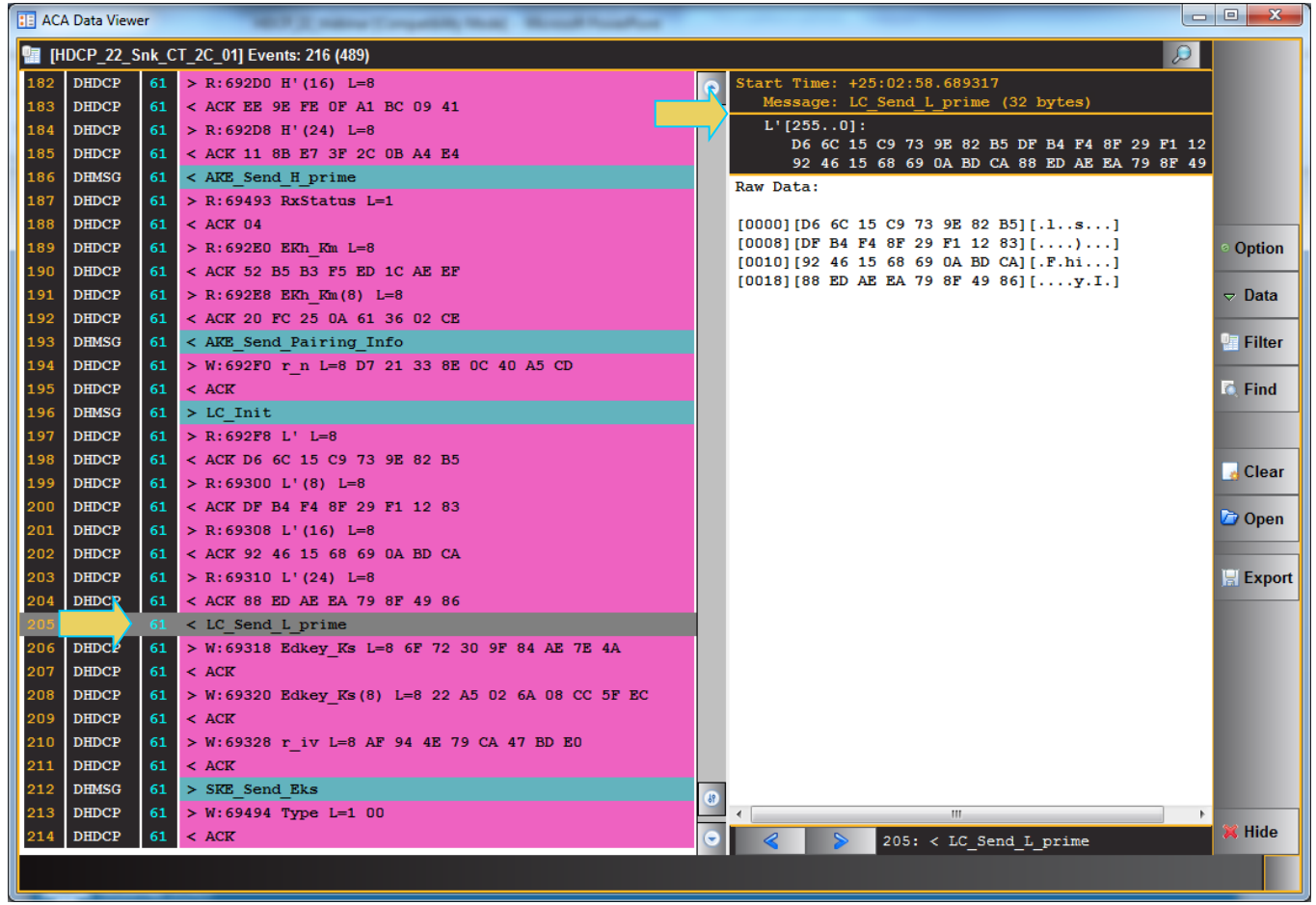




- Transmitter initiates Locality Check.
- Transmitter sends a random number (Rn) to the Receiver.
- Transmitter sets a timer.

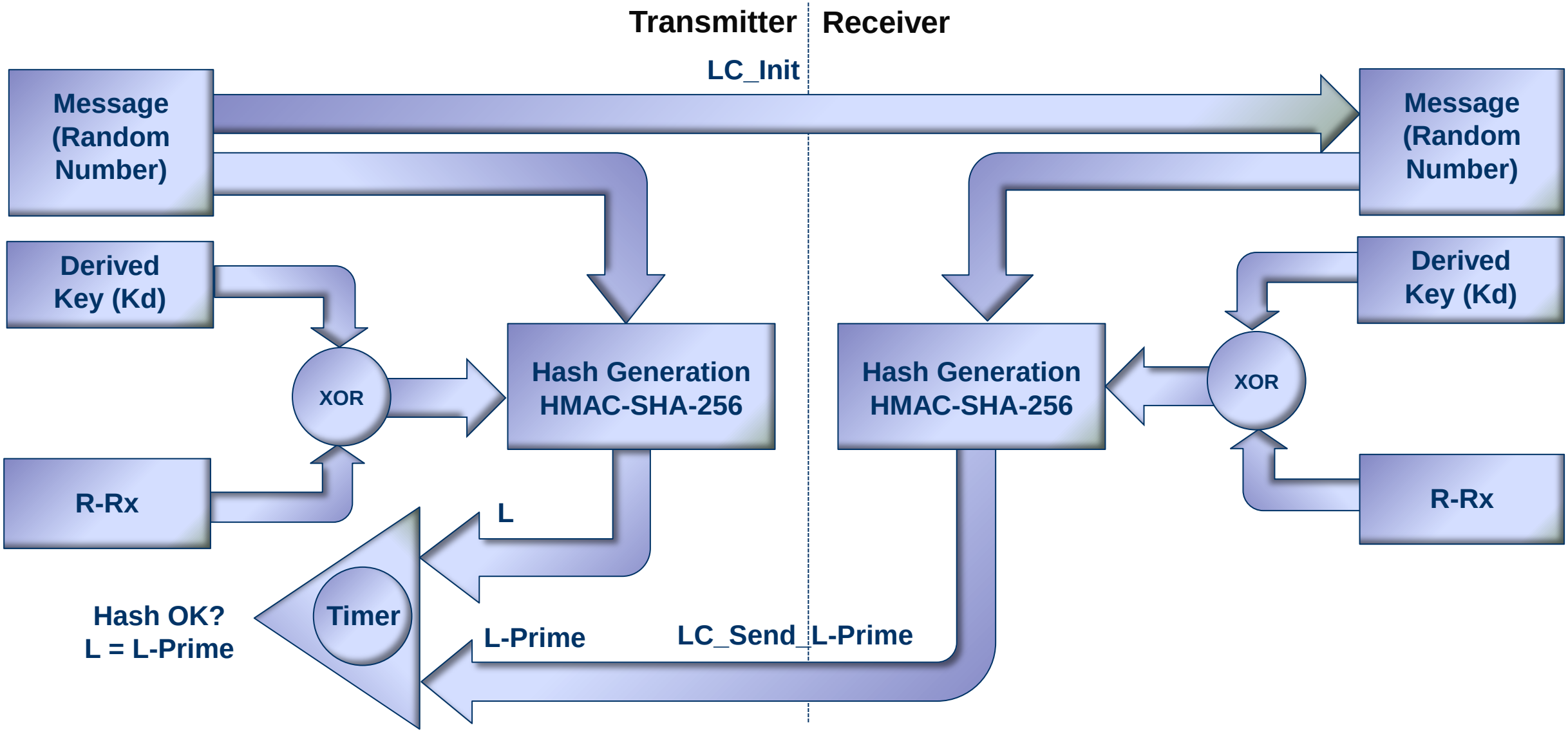
HDCP 2.2 Sequence – Transmitter Verifies Locality





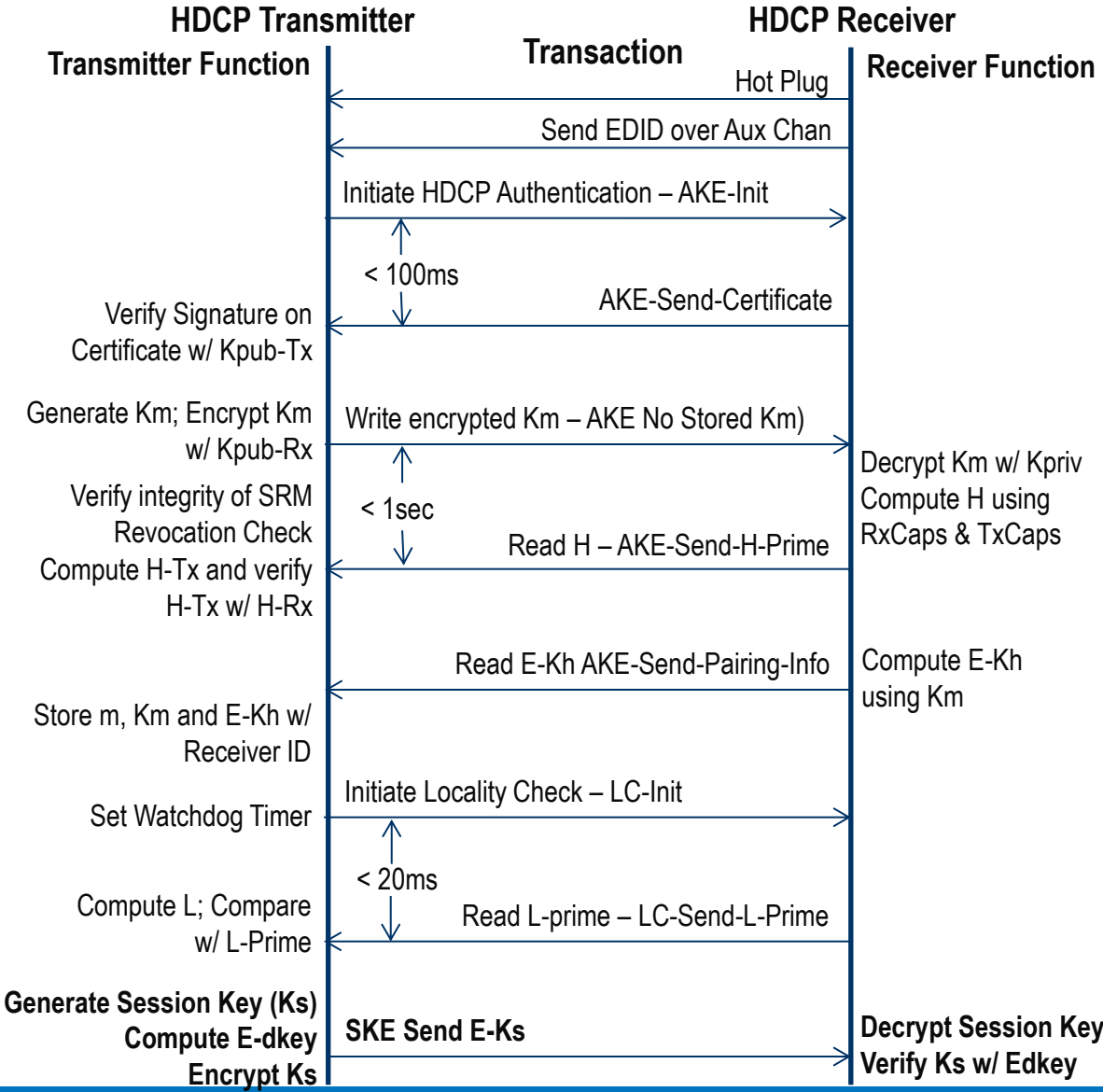
- Transmitter checks for an L-Prime (256-bit number) by comparing it to its L value (256-bit number).
- Verifies that L Prime is return within 20 msec.
- L and L-Prime are generated from the Rn with Hash SHA-256.

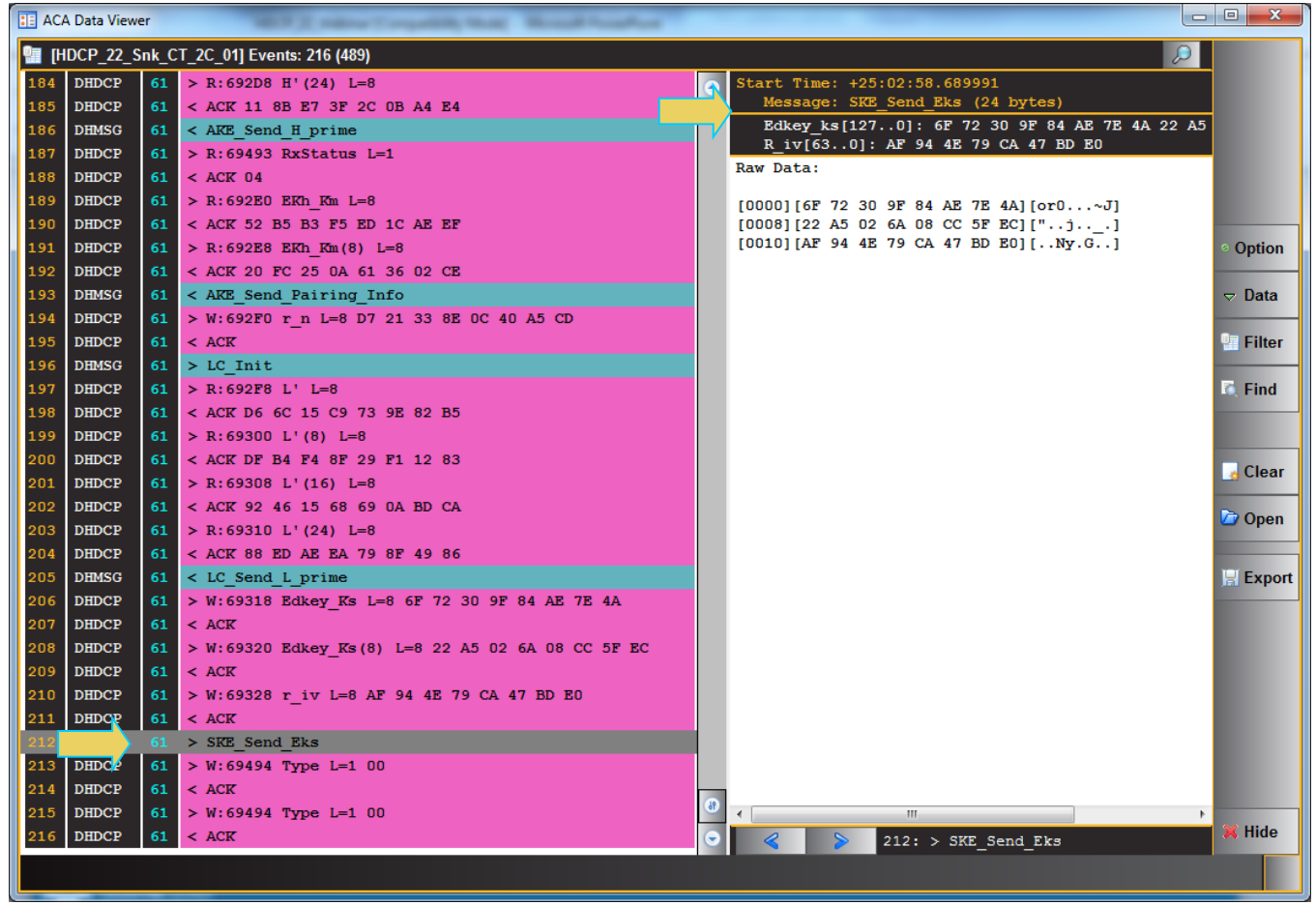
HDCP 2.2 Sequence – Transmitter Verifies Locality



HDCP Session Key Exchange

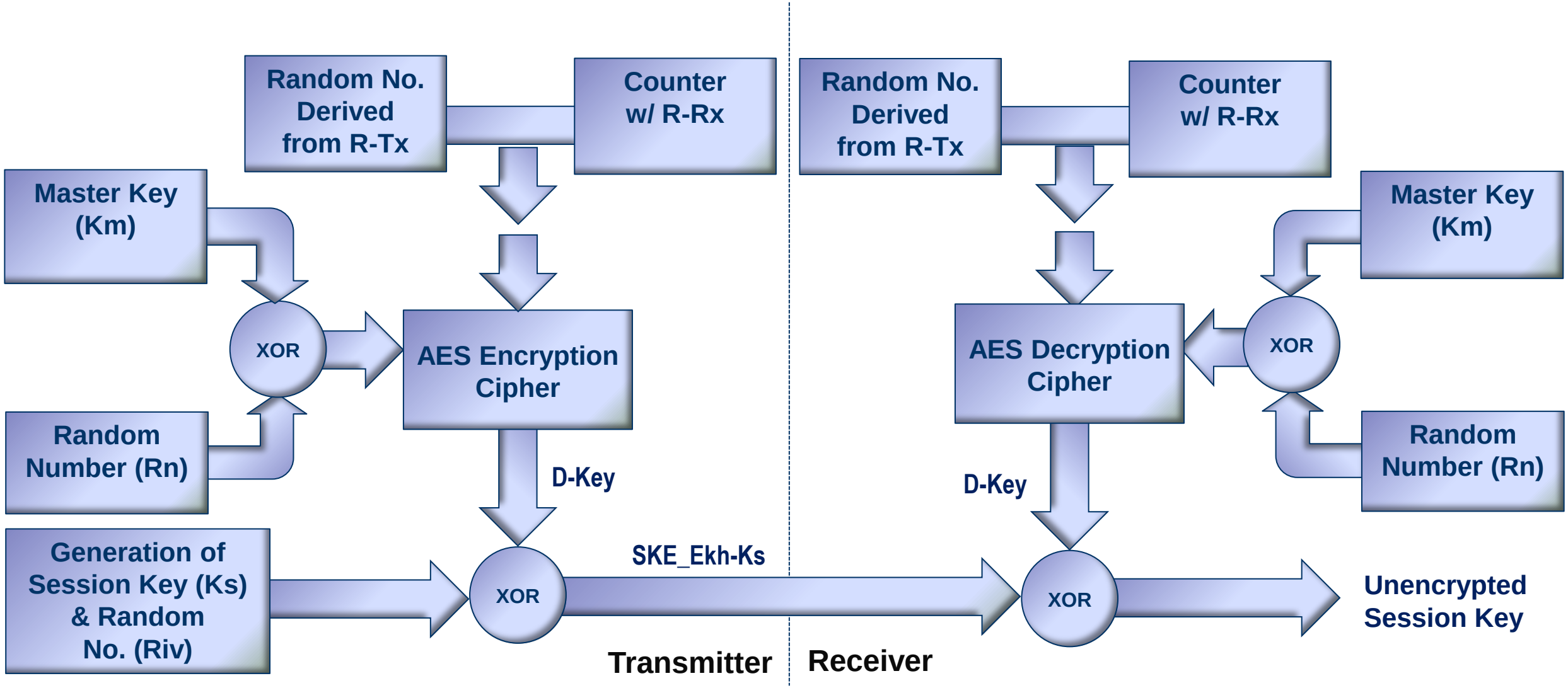
HDCP 2.2 Sequence – Session Key Generation and Exchange





- Transmitter generates random Session Key (Ks).
- The Session Key is AES-encrypted using Master Key (Km)
- Transmitter sends Session Key (Km) to Receiver with a pseudo-random number (Riv) in SKE E-Ks write message.

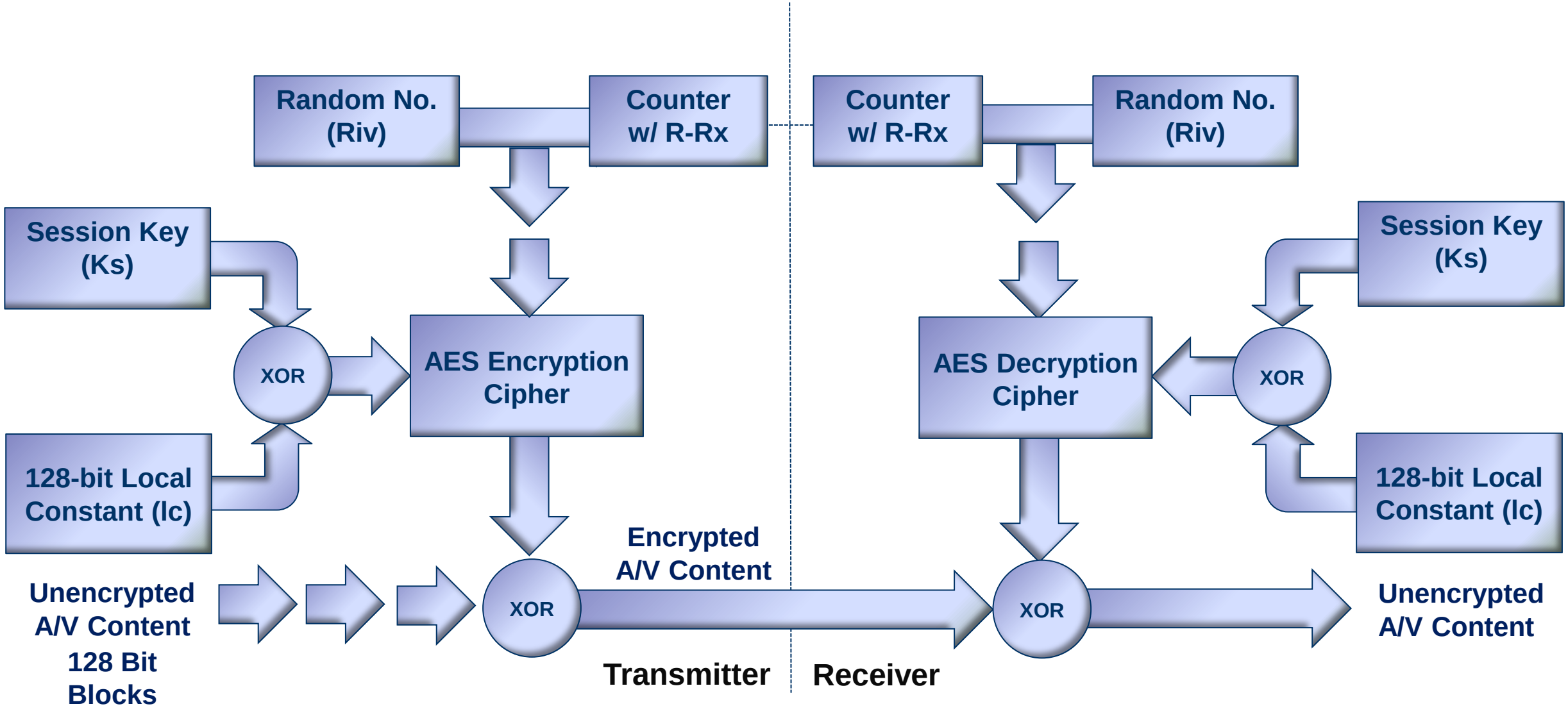
HDCP 2.2 – Session Key Generation, Encryption and Exchange



HDCP 2.2 Encryption

TELEDYNE LECROY
Everywhereyoulook™

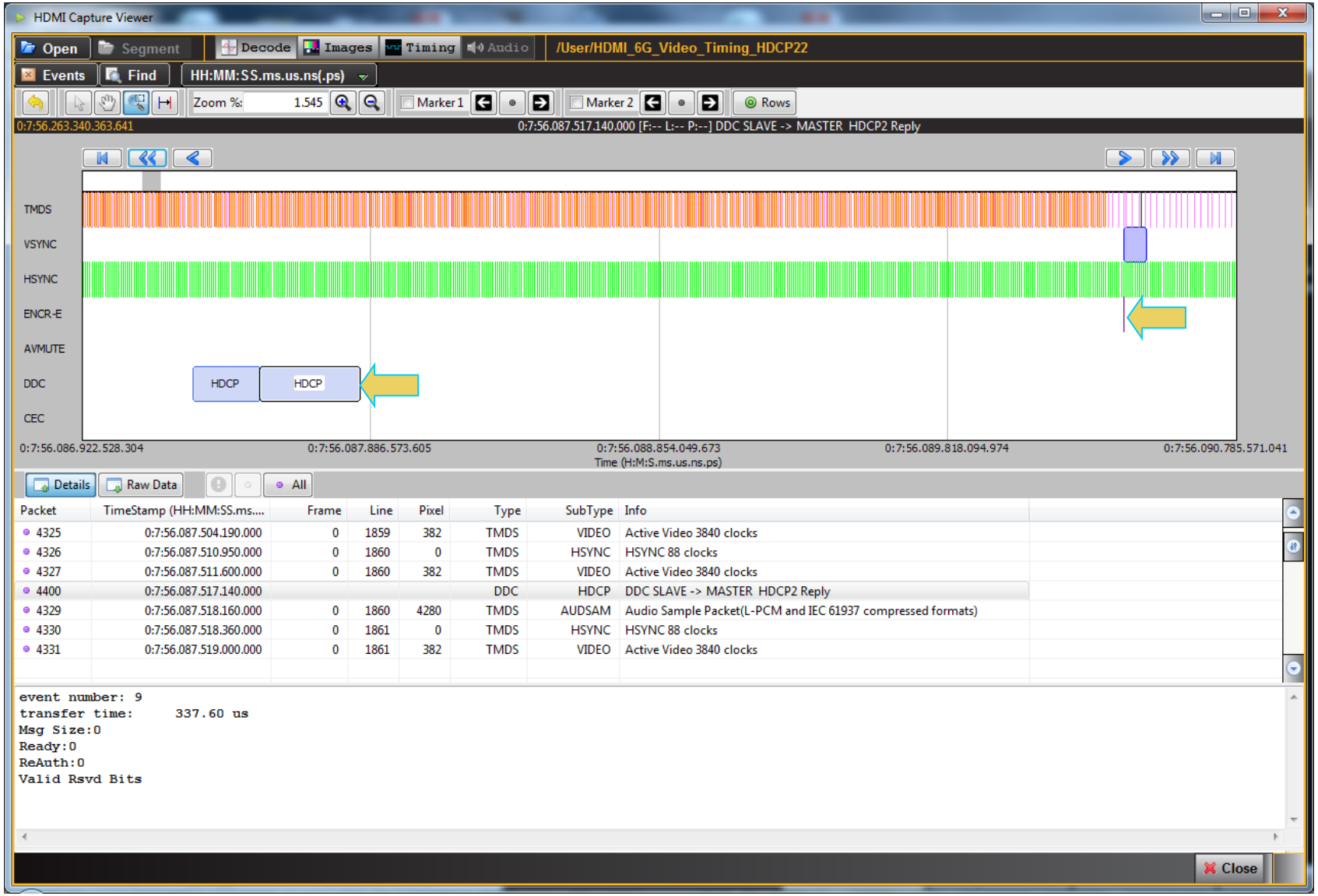
HDCP 2.2 – A/V Content Encryption



HDCP Encryption Status Signaling (HDMI)

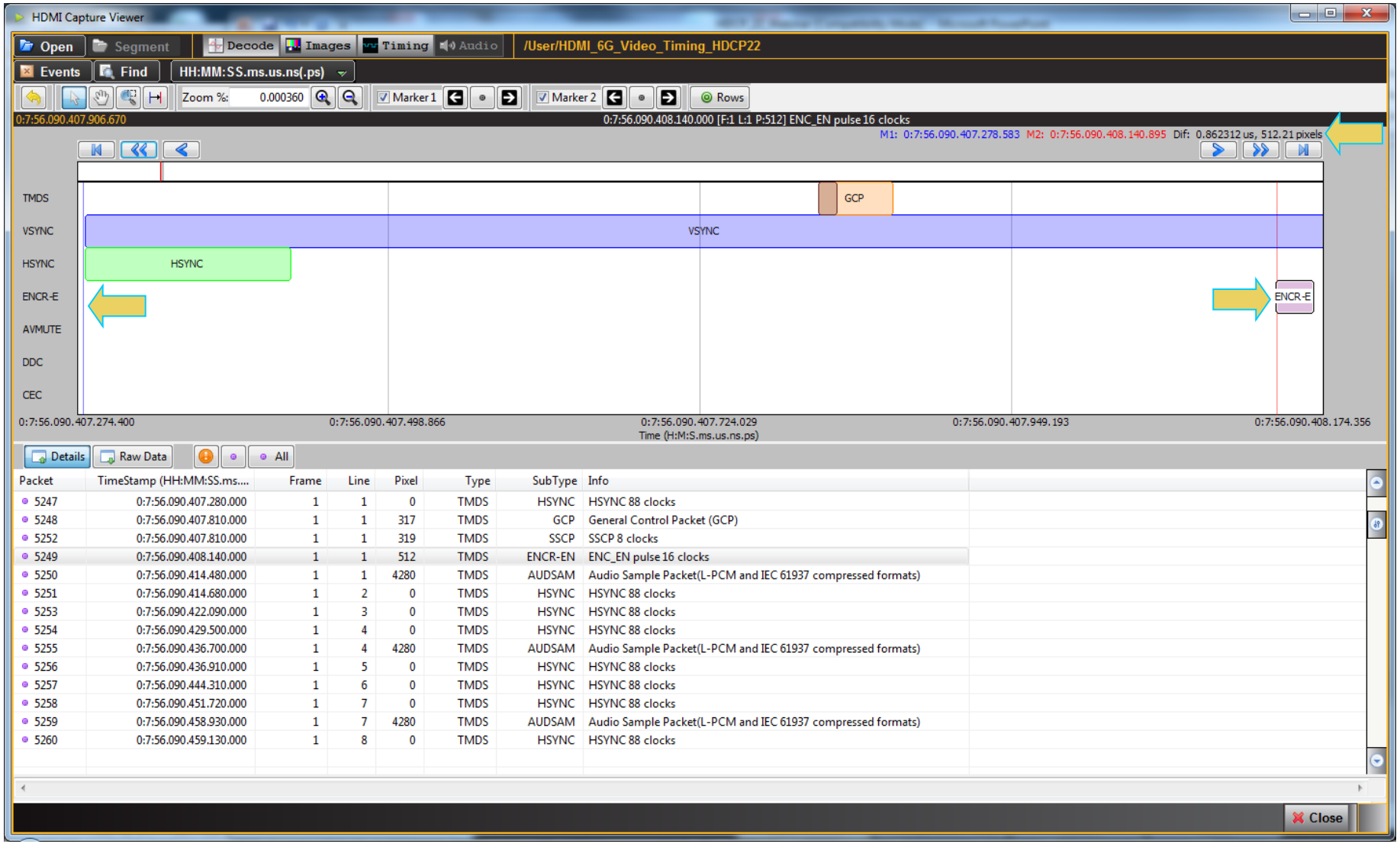
- HDCP Transmitter indicates to the Receiver to begin decrypting of the HDCP protected incoming stream using the Encryption Control Signals (CTL3, CTL2, CTL1, CTL0).
  - These signals (Encryption Enable Pulse) must be transmitted within a 16-clock “Window of Opportunity” starting at 512 pixel clocks following the active edge of Vsync.
- “Keep out Period” (Applies to HDMI)
  - No Data Islands (e.g. InfoFrames, Audio, Control Data) or Video data should be transmitted during this “Keep out Period” starting from 508th pixel after the active edge of Vsync and continuing to 650 pixels after the active edge of Vsync.

HDCP 2.2 Encryption Enable Pulse (HDMI)



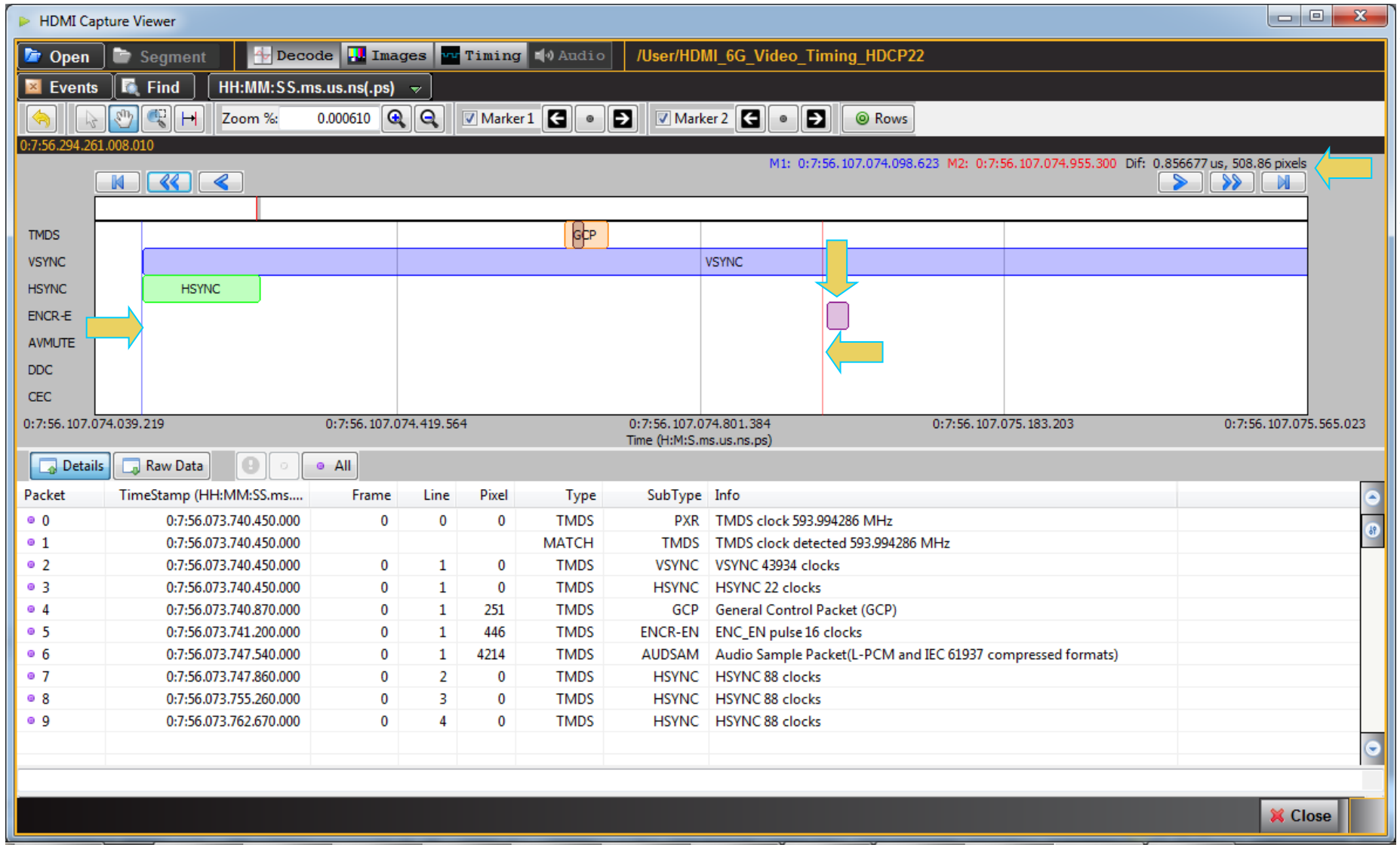
- HDMI-HDCP stream indicating Video, Encryption Enable Pulse and HDCP authentication messages.

HDCP 2.2 Encryption Enable Pulse (HDMI)



- HDMI-HDCP stream indicating Video, Encryption Enable Pulse and HDCP authentication messages.
- Encryption Enable pulse has to occur 512 pixels following active edge of Vsync.

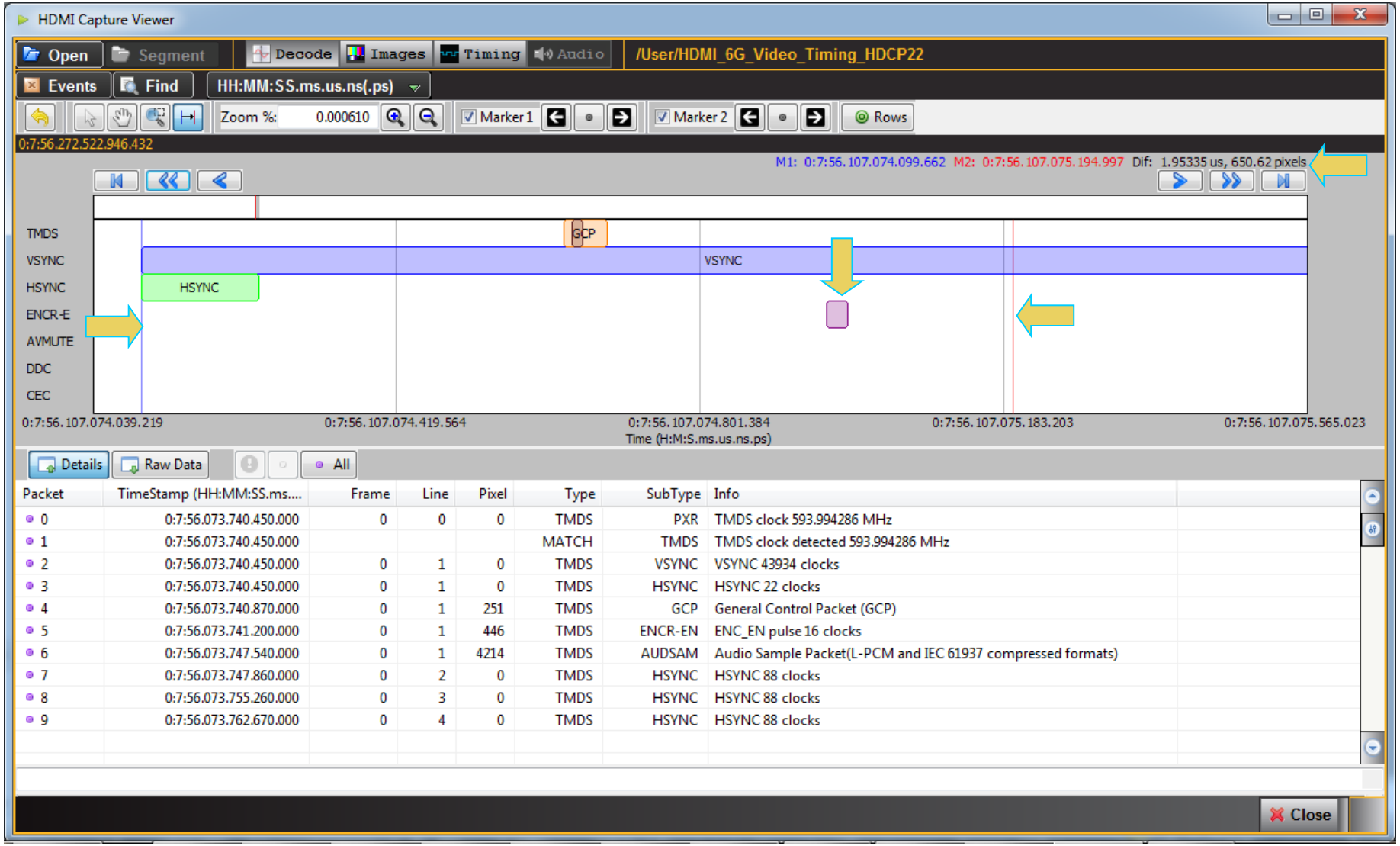
HDCP 2.2 Keep-out Region (HDMI)



Packet	TimeStamp (HH:MM:SS.ms....	Frame	Line	Pixel	Type	SubType	Info
0	0:7:56.073.740.450.000	0	0	0	TMDS	PXR	TMDS clock 593.994286 MHz
1	0:7:56.073.740.450.000				MATCH	TMDS	TMDS clock detected 593.994286 MHz
2	0:7:56.073.740.450.000	0	1	0	TMDS	VSYNC	VSYNC 43934 clocks
3	0:7:56.073.740.450.000	0	1	0	TMDS	HSYNC	HSYNC 22 clocks
4	0:7:56.073.740.870.000	0	1	251	TMDS	GCP	General Control Packet (GCP)
5	0:7:56.073.741.200.000	0	1	446	TMDS	ENCR-EN	ENC_EN pulse 16 clocks
6	0:7:56.073.747.540.000	0	1	4214	TMDS	AUDSAM	Audio Sample Packet(L-PCM and IEC 61937 compressed formats)
7	0:7:56.073.747.860.000	0	2	0	TMDS	HSYNC	HSYNC 88 clocks
8	0:7:56.073.755.260.000	0	3	0	TMDS	HSYNC	HSYNC 88 clocks
9	0:7:56.073.762.670.000	0	4	0	TMDS	HSYNC	HSYNC 88 clocks

- HDMI-HDCP Keep-out region beginning 508 pixels following active edge of Vsync.
- No data islands are supposed to be occurring after that time to the 650th pixel following active edge of Vsync.

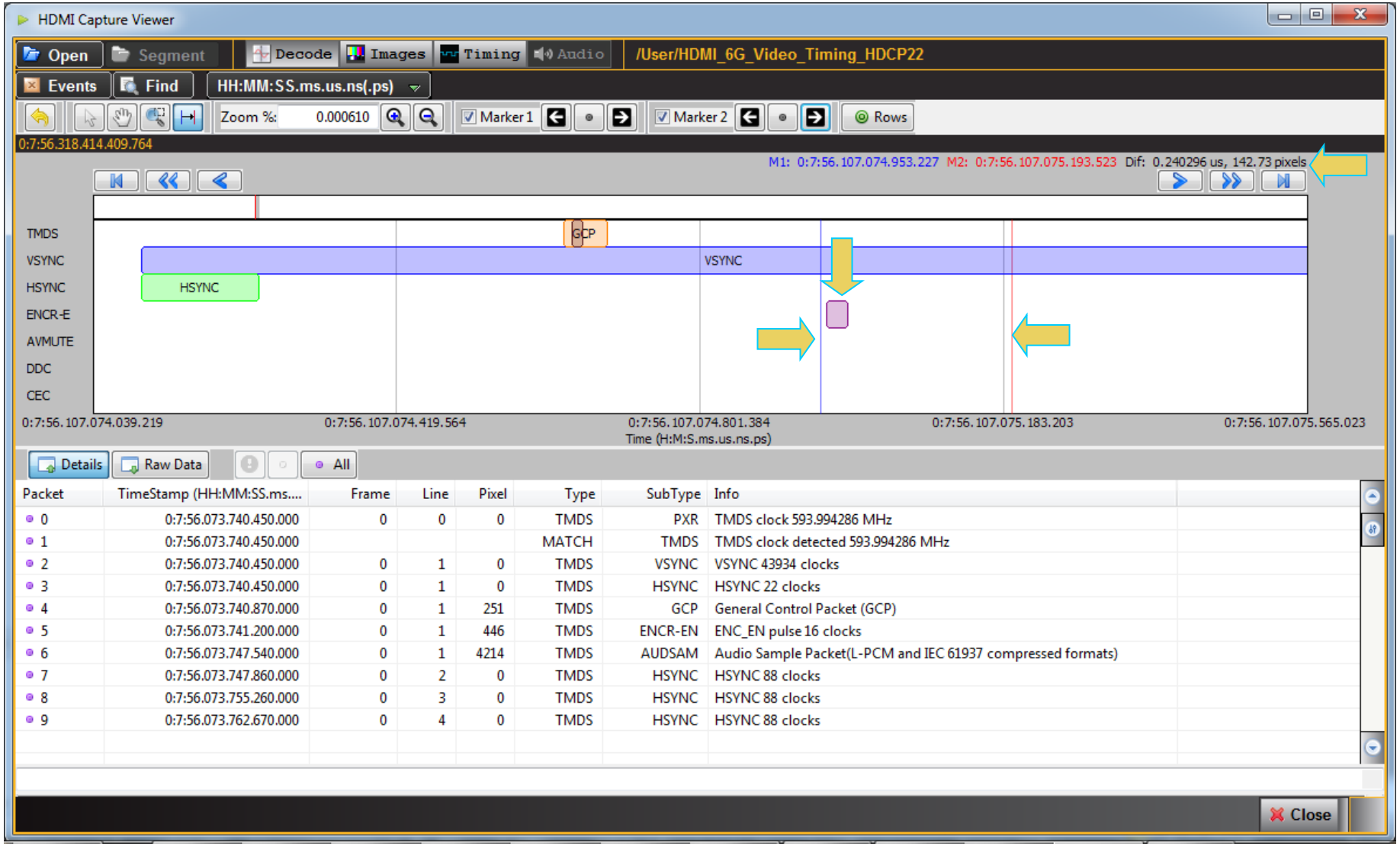
HDCP 2.2 Keep-out Region (HDMI)



Packet	TimeStamp (HH:MM:SS.ms....	Frame	Line	Pixel	Type	SubType	Info
0	0:7:56.073.740.450.000	0	0	0	TMDS	PXR	TMDS clock 593.994286 MHz
1	0:7:56.073.740.450.000				MATCH	TMDS	TMDS clock detected 593.994286 MHz
2	0:7:56.073.740.450.000	0	1	0	TMDS	VSYNC	VSYNC 43934 clocks
3	0:7:56.073.740.450.000	0	1	0	TMDS	HSYNC	HSYNC 22 clocks
4	0:7:56.073.740.870.000	0	1	251	TMDS	GCP	General Control Packet (GCP)
5	0:7:56.073.741.200.000	0	1	446	TMDS	ENCR-EN	ENC_EN pulse 16 clocks
6	0:7:56.073.747.540.000	0	1	4214	TMDS	AUDSAM	Audio Sample Packet(L-PCM and IEC 61937 compressed formats)
7	0:7:56.073.747.860.000	0	2	0	TMDS	HSYNC	HSYNC 88 clocks
8	0:7:56.073.755.260.000	0	3	0	TMDS	HSYNC	HSYNC 88 clocks
9	0:7:56.073.762.670.000	0	4	0	TMDS	HSYNC	HSYNC 88 clocks

- HDMI-HDCP Keep-out region beginning 508 pixels following active edge of Vsync.
- No data islands are supposed to be occurring after that time to the 650th pixel following active edge of Vsync.

HDCP 2.2 Keep-out Region (HDMI)



Packet	TimeStamp (HH:MM:SS.ms....	Frame	Line	Pixel	Type	SubType	Info
0	0:7:56.073.740.450.000	0	0	0	TMDS	PXR	TMDS clock 593.994286 MHz
1	0:7:56.073.740.450.000				MATCH	TMDS	TMDS clock detected 593.994286 MHz
2	0:7:56.073.740.450.000	0	1	0	TMDS	VSYNC	VSYNC 43934 clocks
3	0:7:56.073.740.450.000	0	1	0	TMDS	HSYNC	HSYNC 22 clocks
4	0:7:56.073.740.870.000	0	1	251	TMDS	GCP	General Control Packet (GCP)
5	0:7:56.073.741.200.000	0	1	446	TMDS	ENCR-EN	ENC_EN pulse 16 clocks
6	0:7:56.073.747.540.000	0	1	4214	TMDS	AUDSAM	Audio Sample Packet(L-PCM and IEC 61937 compressed formats)
7	0:7:56.073.747.860.000	0	2	0	TMDS	HSYNC	HSYNC 88 clocks
8	0:7:56.073.755.260.000	0	3	0	TMDS	HSYNC	HSYNC 88 clocks
9	0:7:56.073.762.670.000	0	4	0	TMDS	HSYNC	HSYNC 88 clocks

- HDMI-HDCP Keep-out region 142 pixels.
- No Video or Data Islands allowed.

HDCP 2.2 Compliance Tests

TELEDYNE LECROY
Everywhereyoulook™

HDCP Compliance Testing

- Licensing governed by Digital Content Protection LLC (DCP).
- From DCP website:

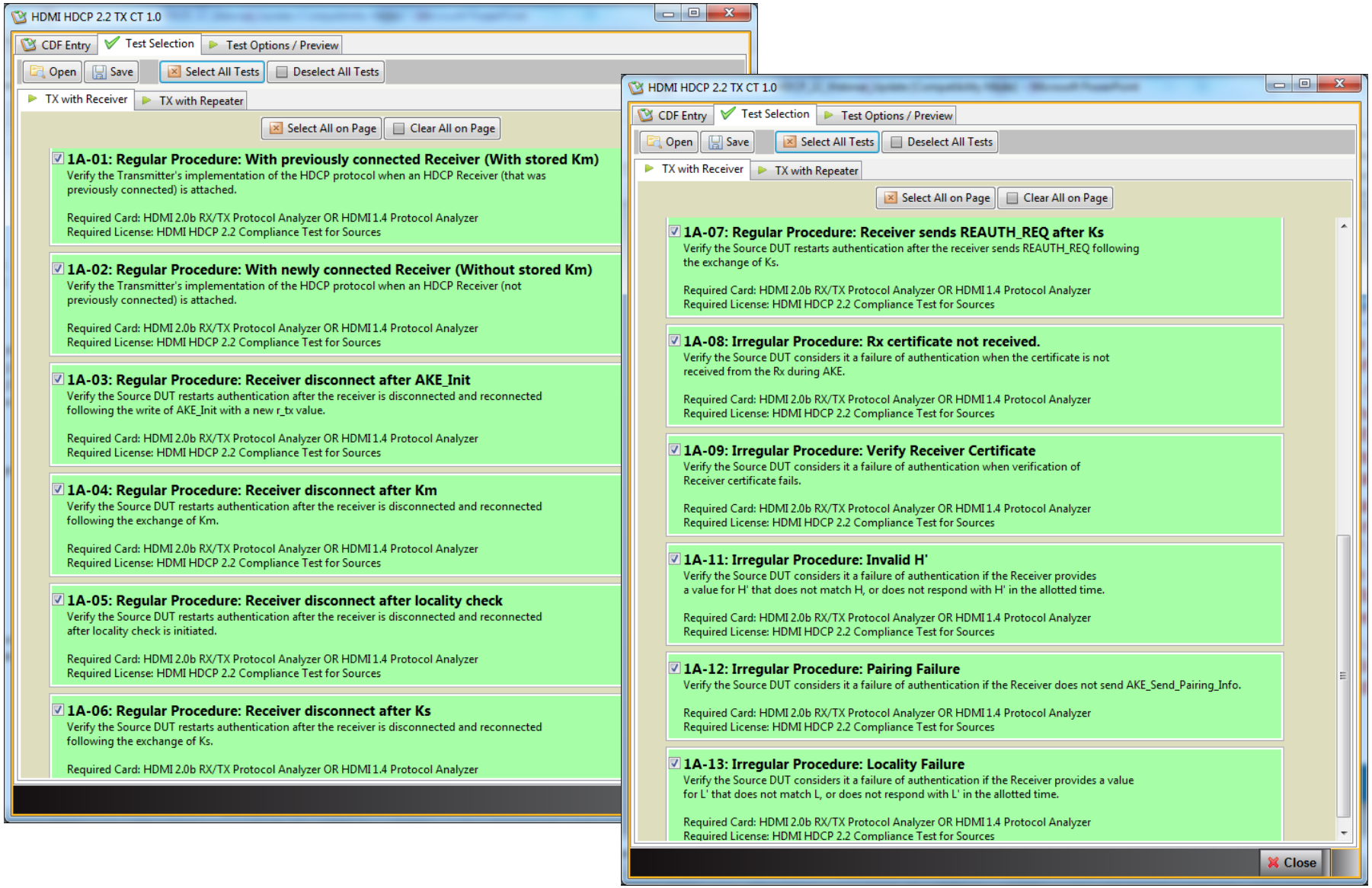
"The Compliance test is intended as an aid to the correct implementation of the Compliance Rules for hardware and software implementations of the HDCP Specification in a Licensed Product. The DCP LLC strongly recommends that you complete this testing for each hardware model or software version of a Licensed Product before releasing any product and at a sufficiently early date in design, as well as during production, to avoid product compliance redesign delays."

Teledyne LeCroy – HDMI, DisplayPort and HDCP Testing

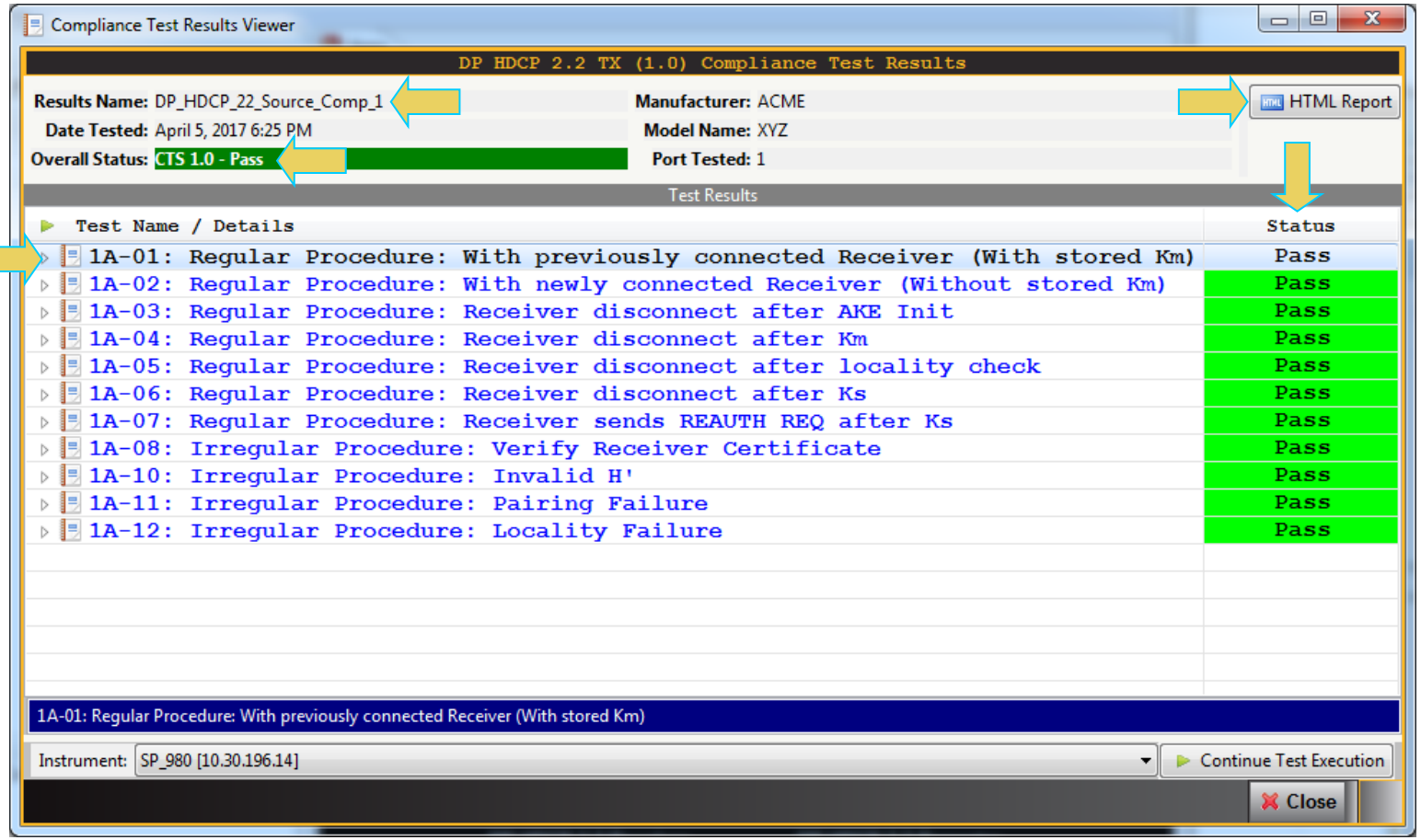
980B Test Platform

Supports all HDCP 2.2 compliance tests, source, sink, repeater for both HDMI and DisplayPort.

HDCP 2.2 Compliance Test – Transmitter 1A Test Series (with Receiver)

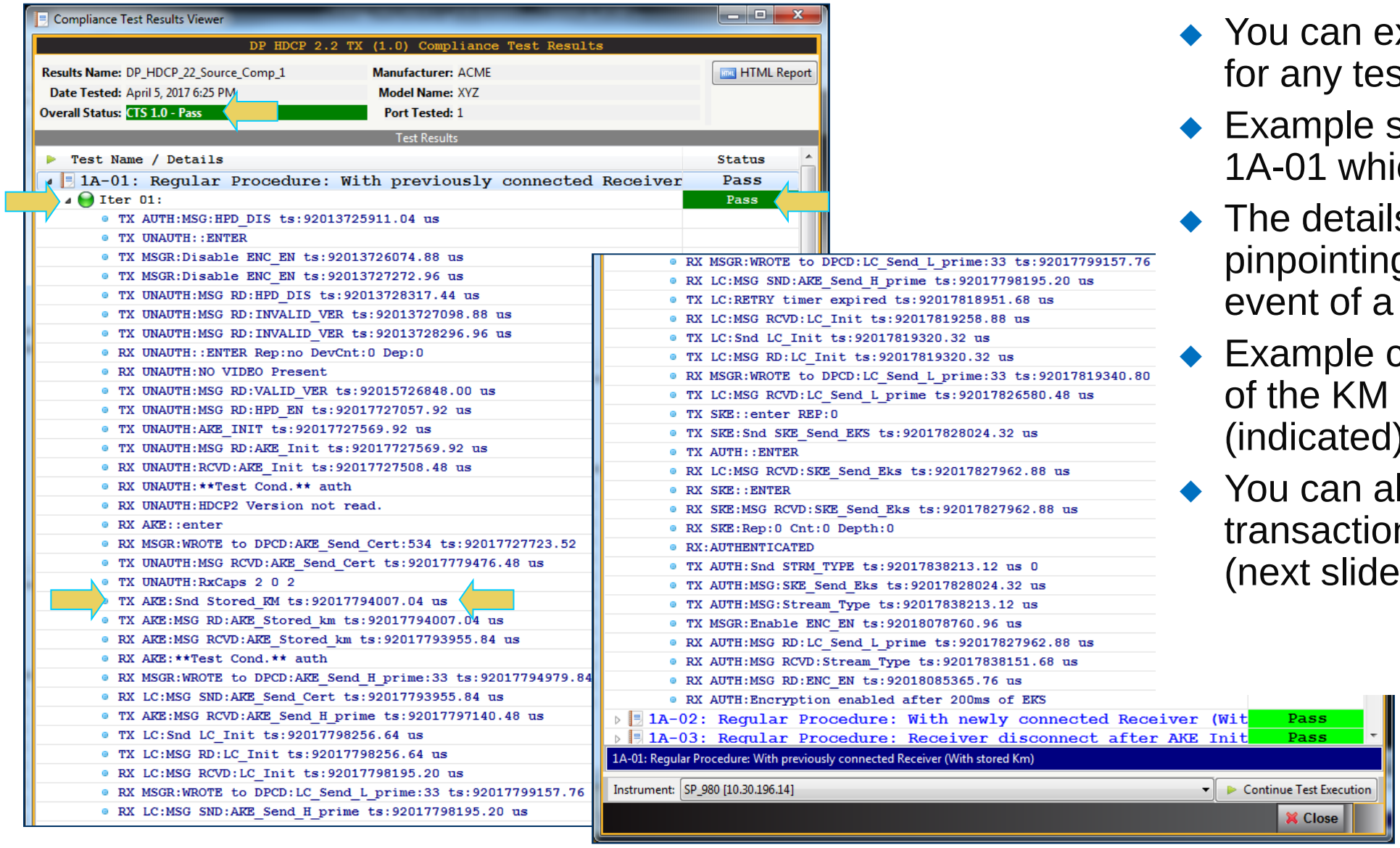


HDCP 2.2 Compliance Test – Test Results Viewer



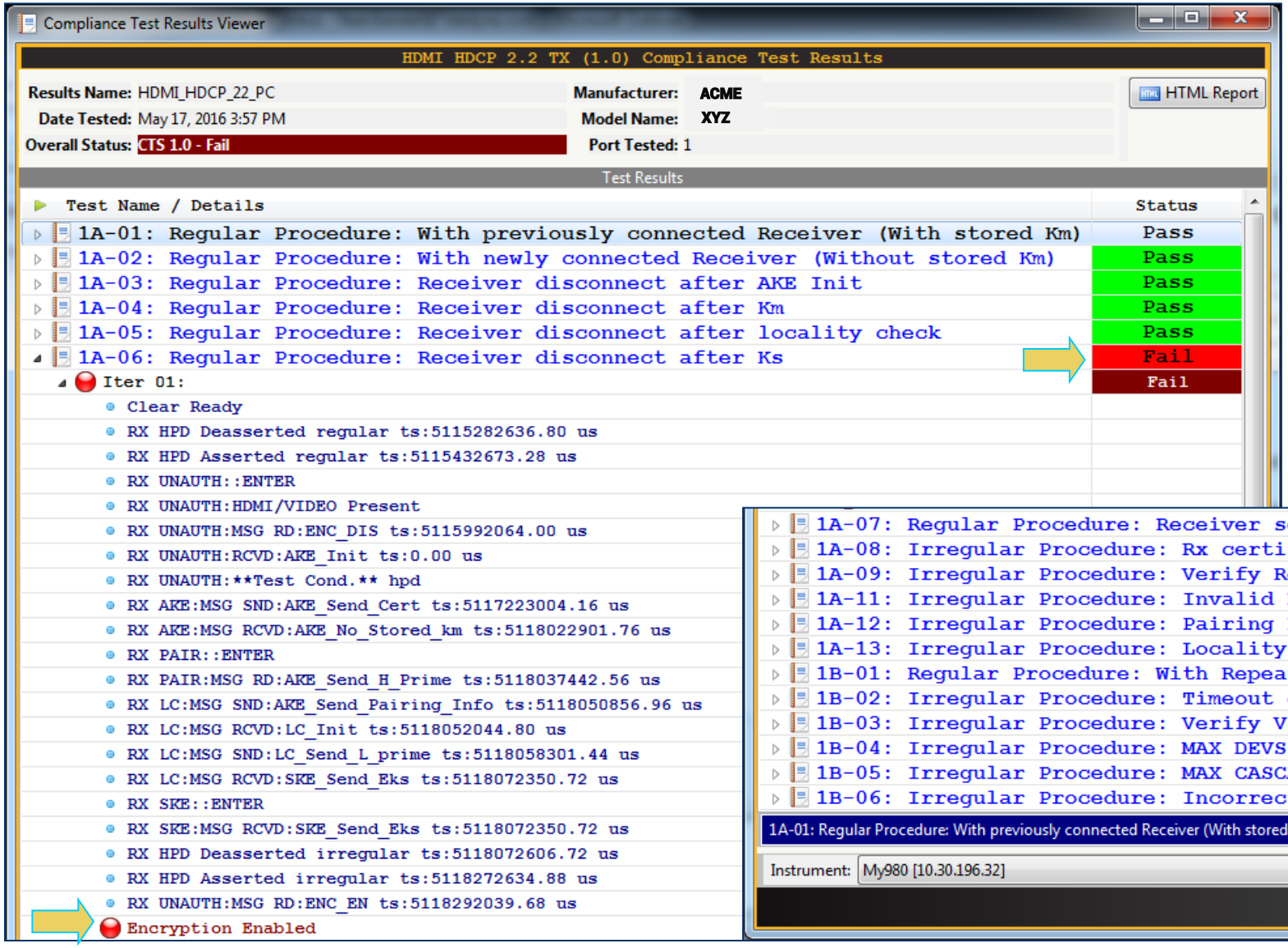
- HDCP 2.2 source Compliance Test for results shows name at the top.
- HDCP 2.2 source Compliance Test results file shows Pass/Fail results of each test, overall Pass/Fail results.
- You can generate an HTML report (button top right).
- We will look at two example tests in the following slides: 1A-01 (a regular test) and 1A-12 tests (irregular test).

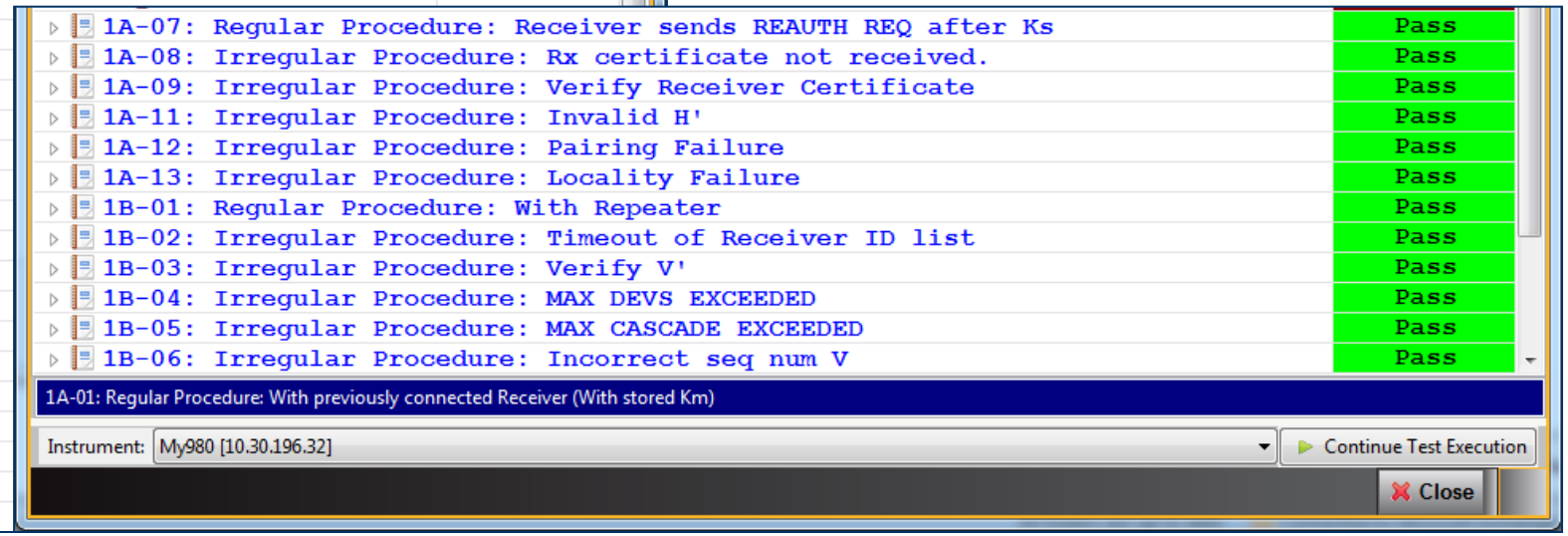
HDCP 2.2 Compliance Test – Example Test 1A-01



- You can explode out the results for any test to view the details.
- Example shows details of test 1A-01 which passes.
- The details are useful for pinpointing the root cause in the event of a failure.
- Example could be the sending of the KM (master key) (indicated).
- You can also view the ACA transactions to confirm a failure (next slide).

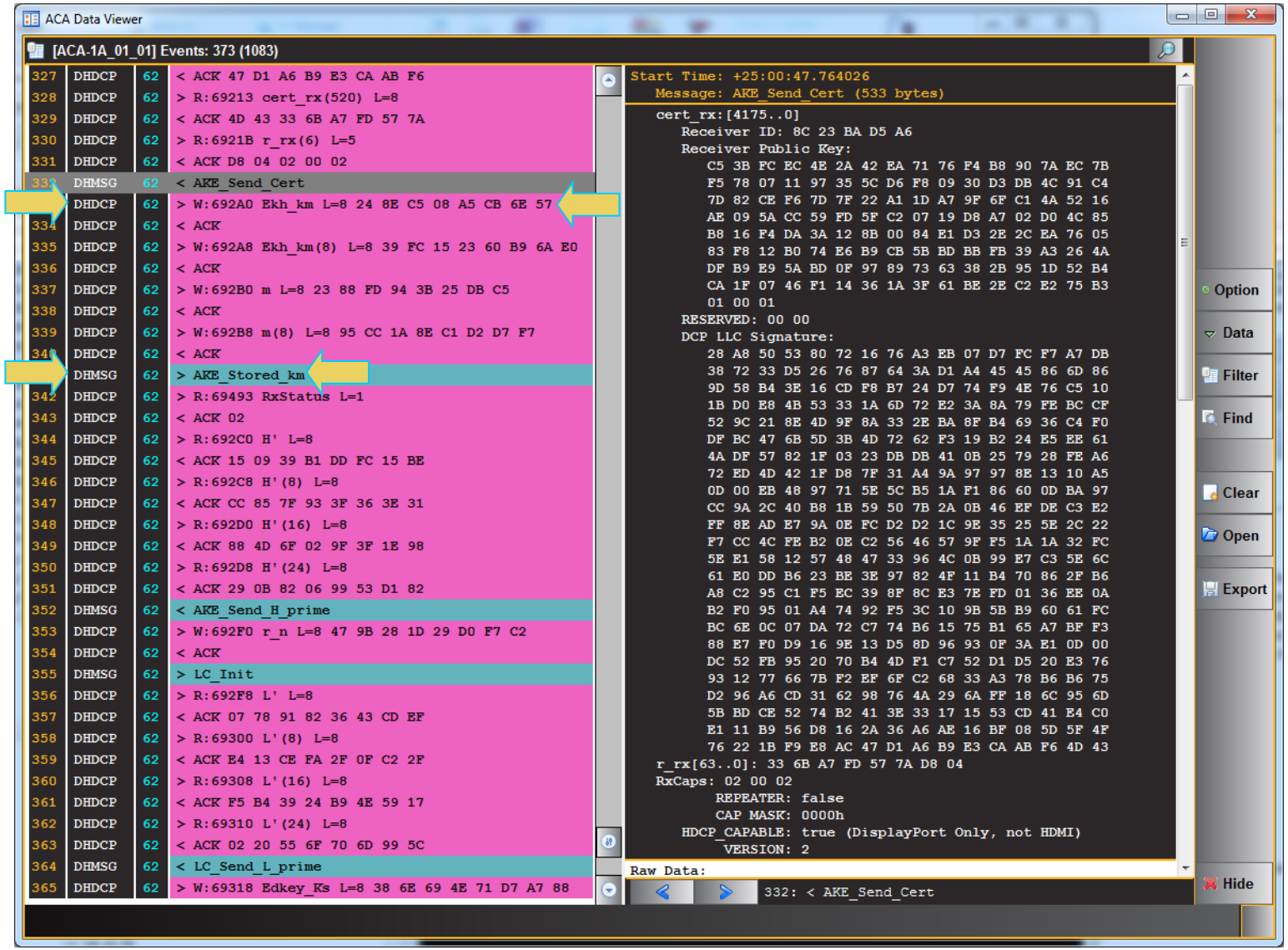
HDCP 2.2 Compliance Test – Example Test 1A-06 Failure





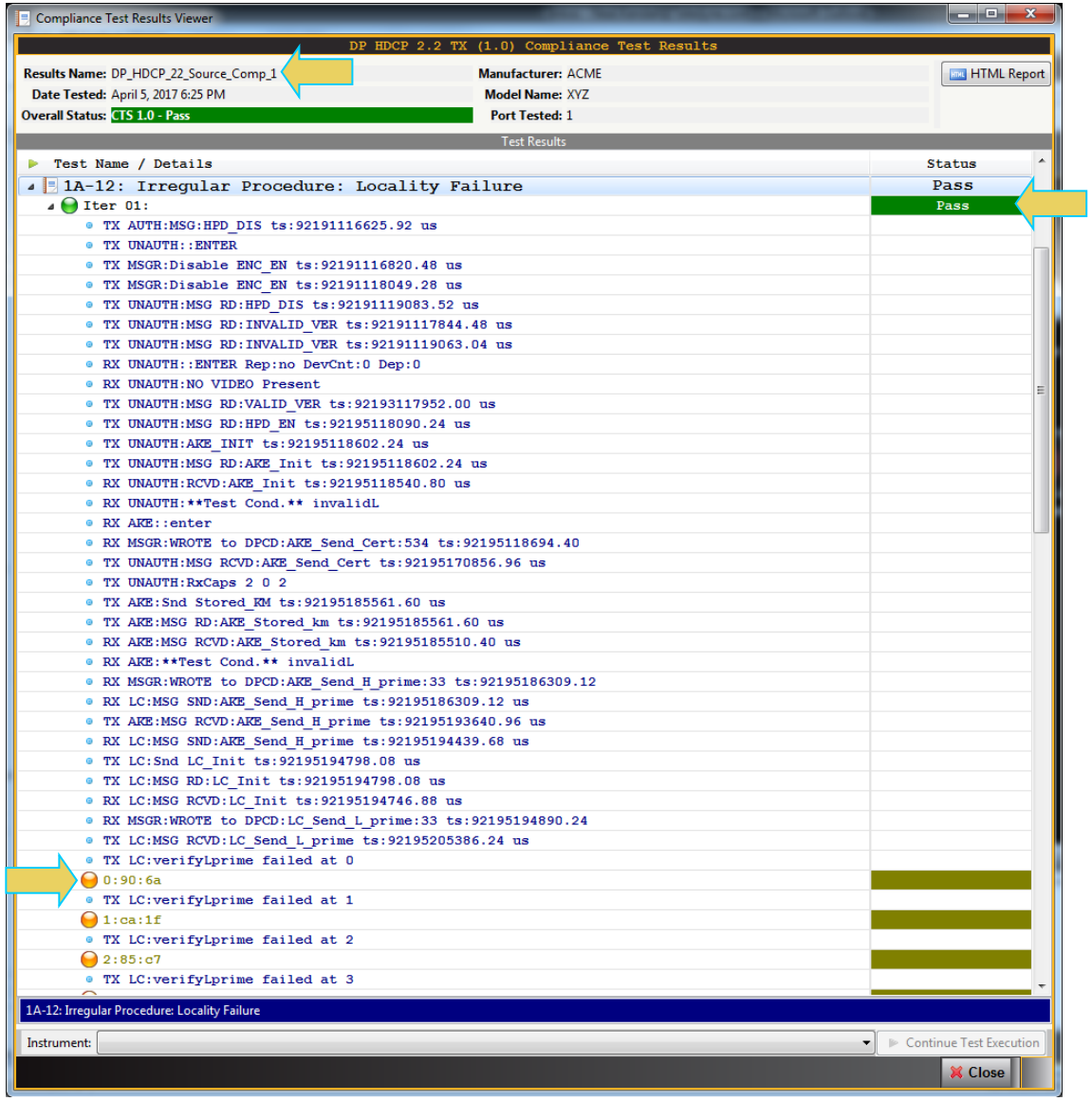
- Example shows details of test 1A-06 where there is a failure.
- The point of failure is identified. In this case, after disconnection cycle, encryption pulse was enabled following a connection cycle. The Transmitter should have terminated the encryption and re-initiated authentication.

HDCP Compliance Testing – HDCP Transactions (Example Test 1A-01)



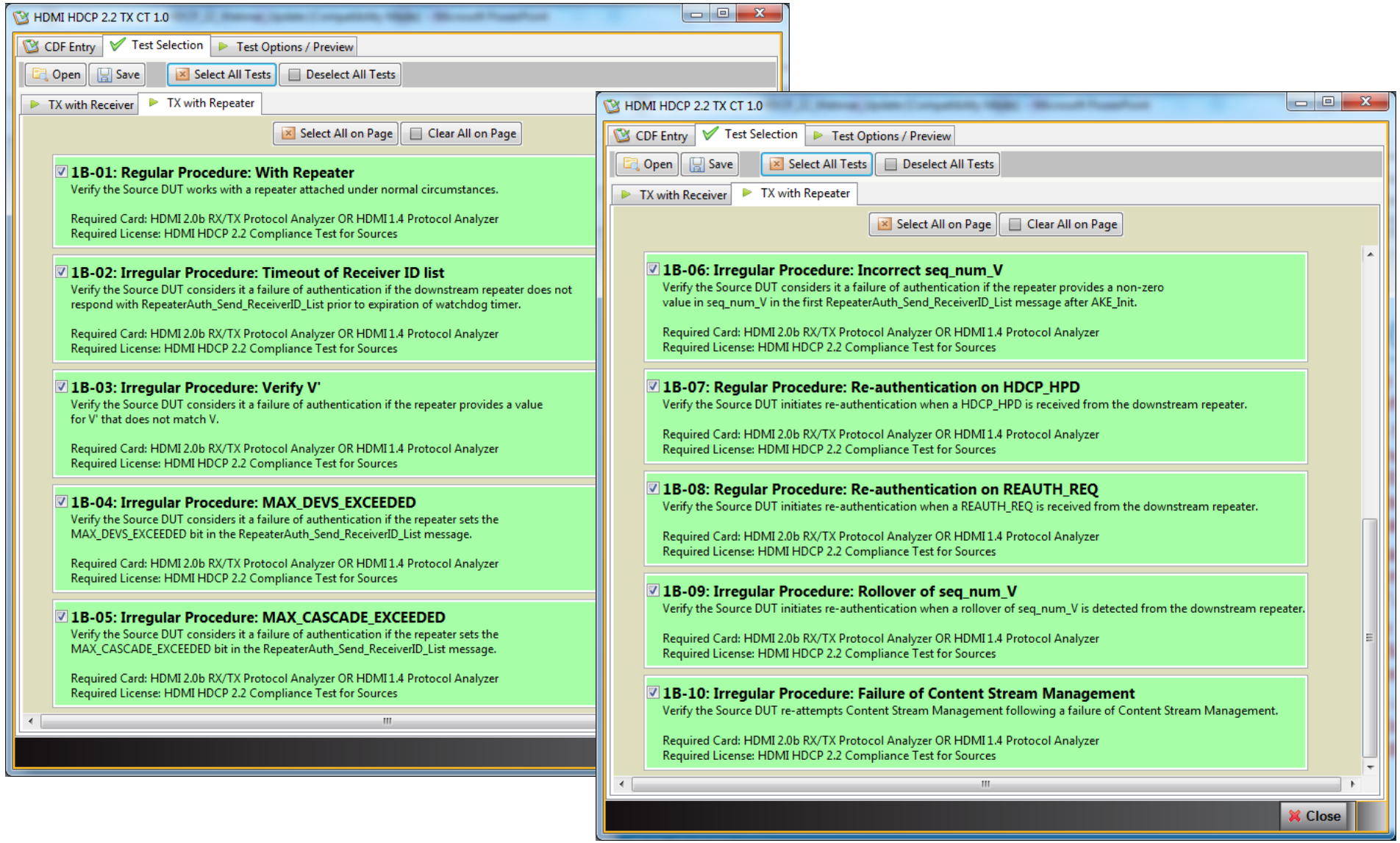
- Example shows details of HDCP transactions occurring over the HDCP 2.2 source compliance test for 1A-01 test.
- Details of selected transaction shown in Details panel.
- Note the related AKE DHDCP messages are consolidated in the DPMSG transactions.

HDCP 2.2 Compliance Test – Test Results Viewer (Example Test 1A-12)

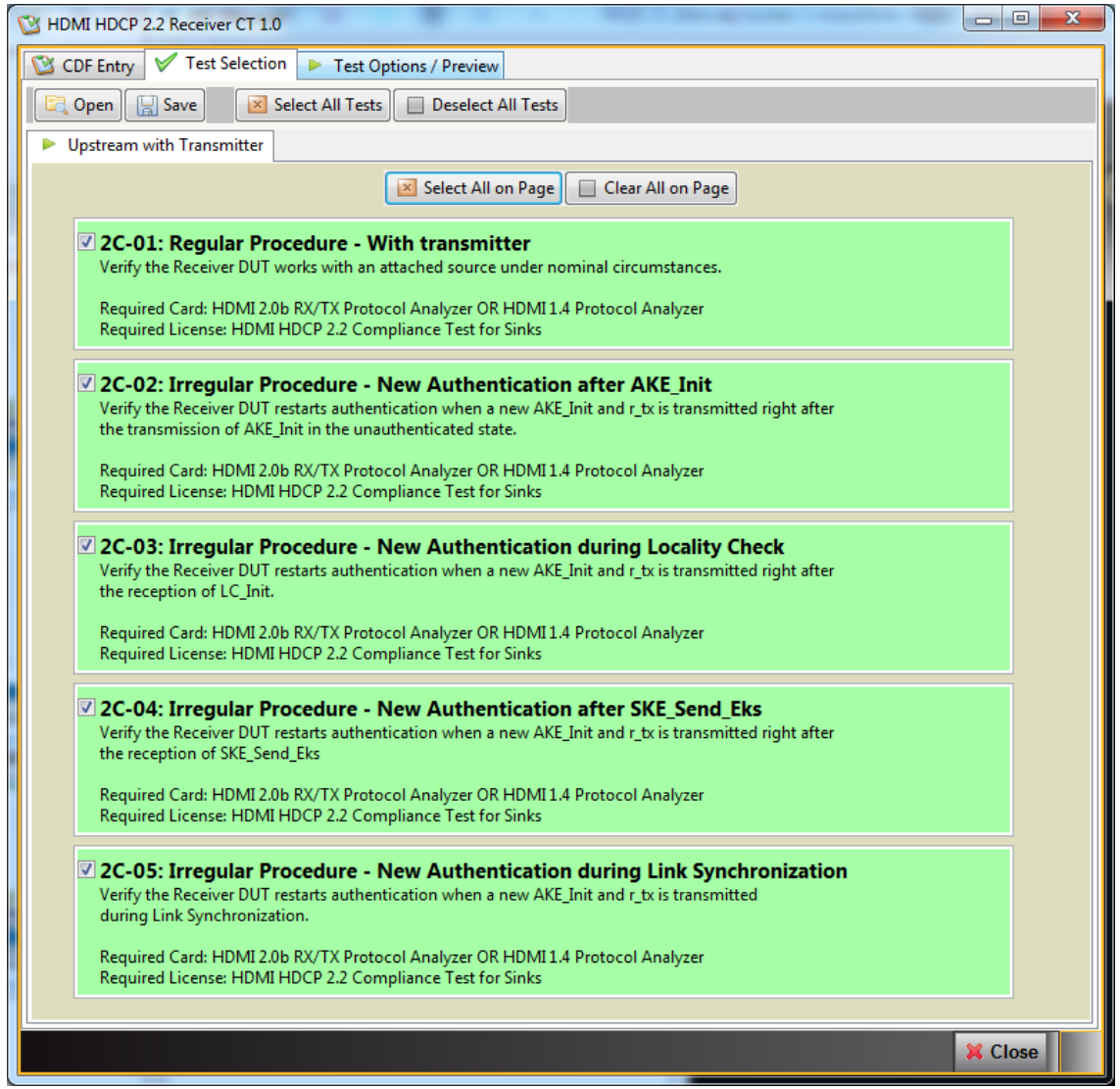


- Example shows details of test 1A-12 which passes.
- Test 1A-12 is an irregular test where the Test Equipment (980 DP/HDMI Protocol Analyzer) responds with an invalid L-Prime value.
- The details are useful for pinpointing the root cause in the event of a failure.

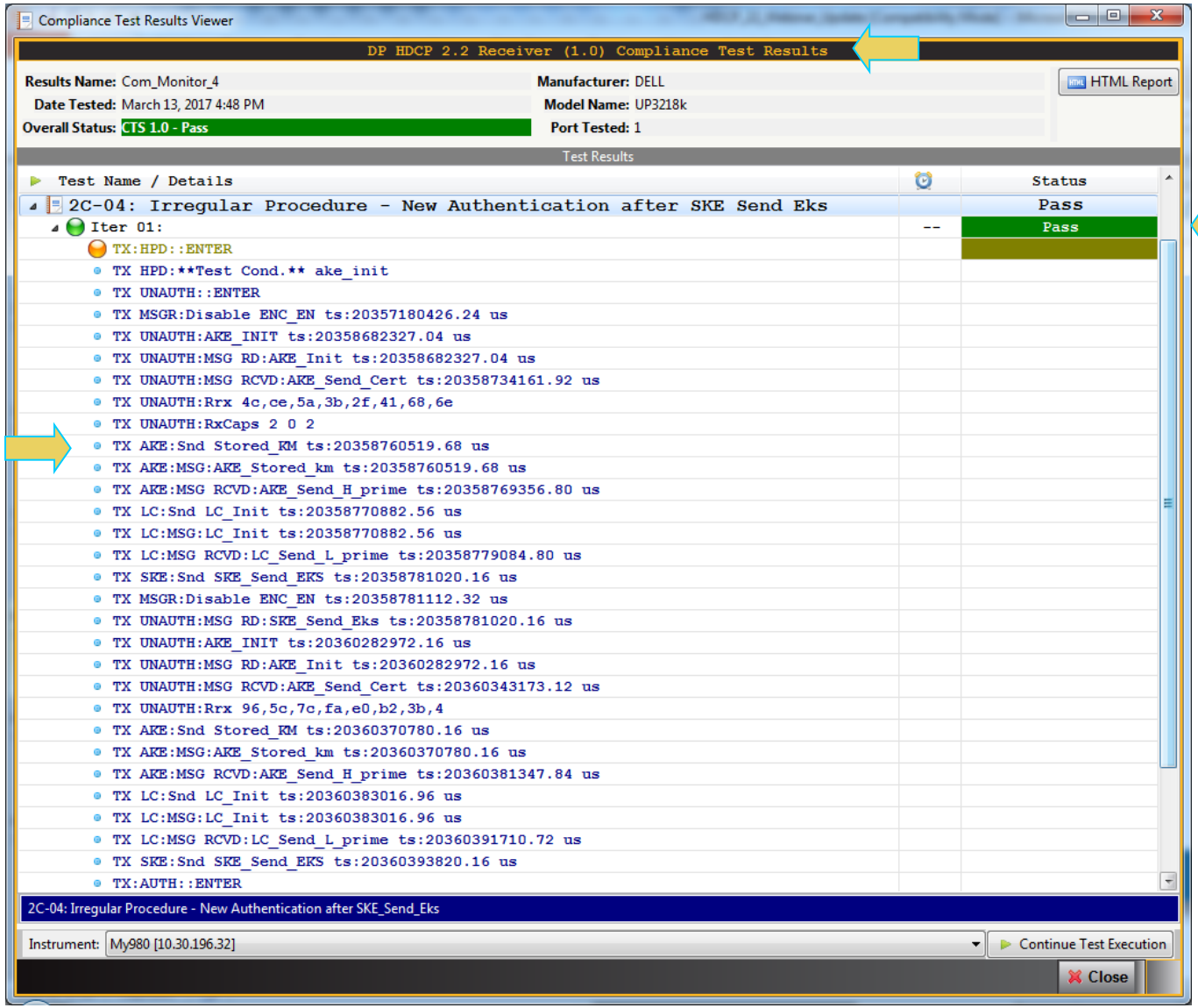
HDCP 2.2 Compliance Test – Transmitter 1B Test Series (with Repeater)



HDCP 2.2 Compliance Test – Receiver 2C Test Series



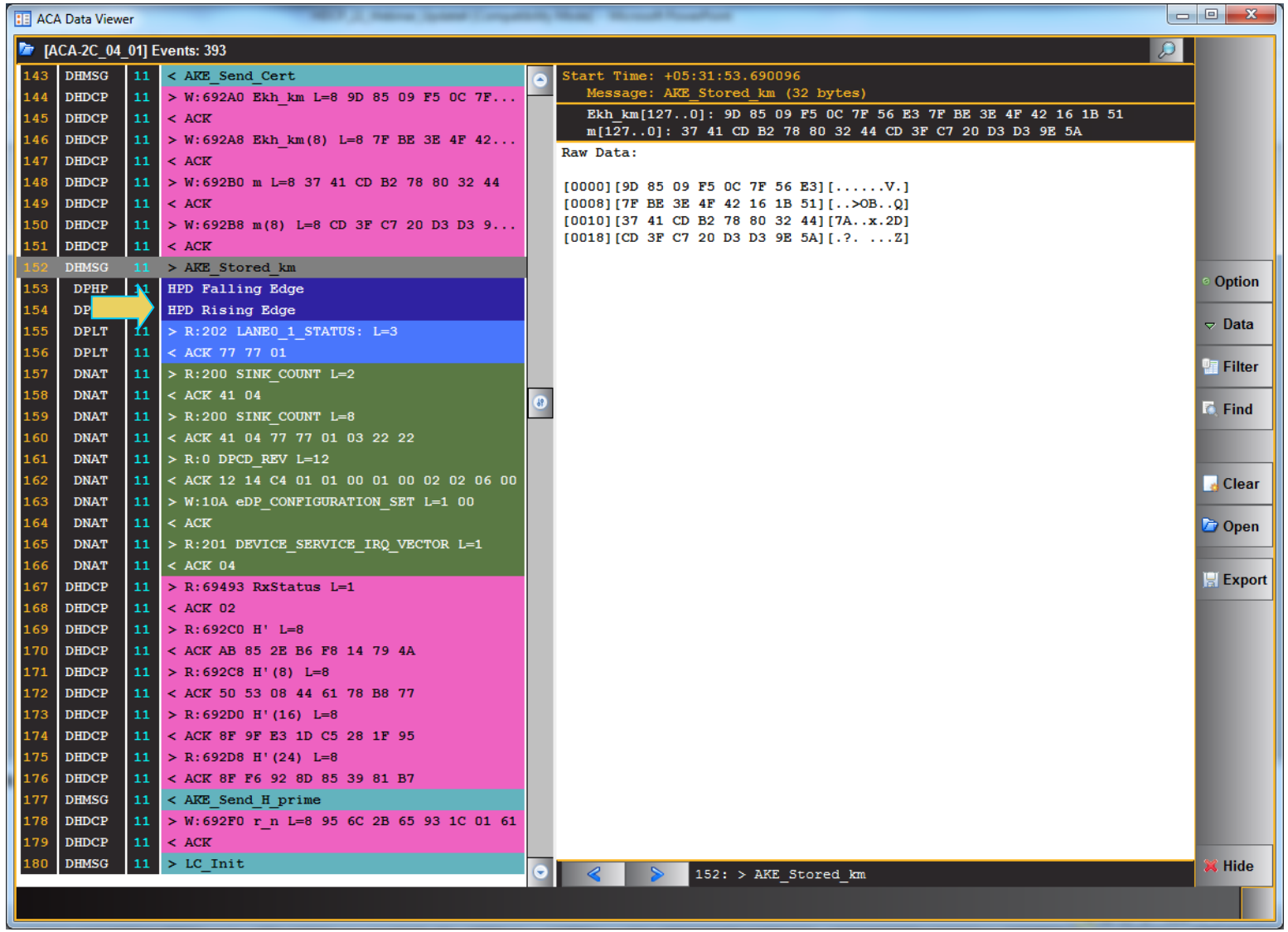
HDCP 2.2 Compliance Test – Test Results Viewer (Example Test 2C-04)



- Example shows details of test 2C-04 which passes.
- Test 2C-04 is an irregular test where the Test Equipment (980 DP/HDMI Protocol Analyzer) is emulating a source that initiates re-authentication during Session Key (Ks) exchange.

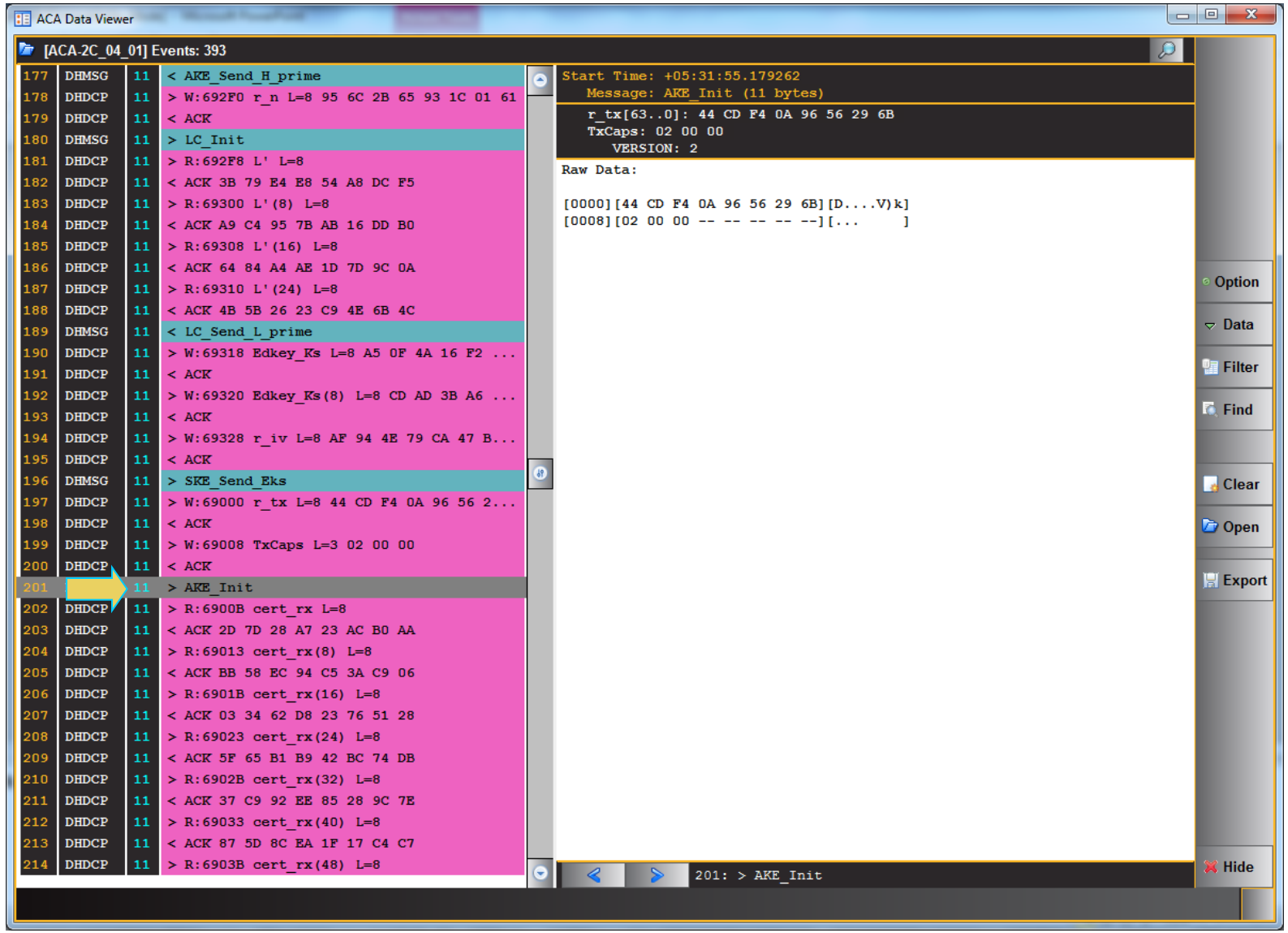
Note: You can also view the ACA transactions to confirm a failure (which we have seen in this webinar already).

HDCP 2.2 Compliance Test – View ACA of Test Results (Example Test 2C-04)



- Example shows details of test 2C-04 which passes.
- Test 2C-04 is an irregular test where the Test Equipment (980 DP/HDMI Protocol Analyzer) is emulating a source that initiates re-authentication during Session Key (Ks) exchange.
- Here you see the disconnect occurring. Immediately following the disconnect the Transmitter reads the Receiver's DPCD registers.
- There is some residual activity of the previous authentication.

HDCP 2.2 Compliance Test – View ACA of Test Results (Example Test 2C-04)



- Here you see the re-initiation of authentication following the disconnect.

Thank you for attending

Questions?

Please take the brief survey that follows.

Please contact me, Neal Kendall at:
neal.kendall@teledyne.com
If you have any questions.

- We will be announcing additional webinars on the following topics in the coming months; possible topics are:
  - HDMI 2.1 Protocols
  - DisplayPort 1.4 Protocols (e.g. DSC/FEC)
  - DisplayPort Multi-Stream Transport (MST)
  - Dynamic High Dynamic Range