IN THE UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF DELAWARE

MEDIA CONTENT PROTECTION LLC,

Plaintiff,

v.

HP INC.,

Defendant.

C.A. No.: 20-cv-1241-CFC

JURY TRIAL DEMANDED

SECOND AMENDED COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff Media Content Protection LLC ("MCP" or "Plaintiff") brings this action for patent infringement under 35 U.S.C. § 271 against HP, Inc. ("HP" or "Defendant"), and alleges as follows:

THE PARTIES

1. Plaintiff Media Content Protection LLC ("MCP") is a limited liability company duly organized and existing under the laws of the State of Delaware with its principal place of business at 533 Congress Street, Portland, ME 04101.

2. Defendant HP Inc. is a corporation duly organized and existing under the laws of the State of Delaware with a principal place of business located at 1501 Page Mill Road, Palo Alto, CA 94304.

3. Defendant makes, uses, sells, offers for sale, and/or imports throughout the United States, including within the District of Delaware (this "District"), products, such as digital video-capable devices and components thereof, that infringe the Asserted Patents, defined below. Defendant orders and purchases components, such as digital video capable integrated circuits and associated firmware, that it incorporates into digital video-capable devices that are made, used, sold, offered for

sale, and/or imported throughout the United States, including within this District. These digital video-capable devices may include, but are not limited to, desktops, laptops, all-in-one- PCs, thin clients, tablets, convertible PCs, workstations, monitors, displays, projectors, video adapters, and/or video hubs.

THE ASSERTED PATENTS

U.S. Patent No. 9,436,809

4. United States Patent No. 9,436,809 (the "'809 Patent") is entitled "Secure Authenticated Distance Measurement" and issued on September 6, 2016 to inventor Franciscus L. A. J. Kamperman. The '809 Patent issued from United States Patent Application No. 14/538,493 filed on November 11, 2014. A copy of the '809 Patent is attached hereto as Exhibit A.

U.S. Patent No. 10,091,186

5. United States Patent No. 10,091,186 (the "'186 Patent") is entitled "Secure Authenticated Distance Measurement" and issued on October 2, 2018 to inventor Franciscus L. A. J. Kamperman. The '186 Patent issued from United States Patent Application No. 15/352,646 filed on November 16, 2016. A copy of the '186 Patent is attached hereto as Exhibit B.

U.S. Patent No. 10,298,564

6. United States Patent No. 10,298,564 (the "'564 Patent") is entitled "Secure Authenticated Distance Measurement" and issued on May 21, 2019 to inventor Franciscus L. A. J. Kamperman. The '564 Patent issued from United States Patent Application No. 16/117,019 filed on August 30, 2018. A copy of the '564 Patent is attached hereto as Exhibit C.

7. By way of assignment, MCP owns all rights, title, and interest to the '809 Patent, '186 Patent, and '564 Patent (collectively, the "Asserted Patents").

8. The Asserted Patents are each valid and enforceable.

JURISDICTION AND VENUE

9. This is a civil action for patent infringement arising under the Patent Act, 35 U.S.C. § 1 *et seq*.

10. This Court has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

11. Venue in this District is proper pursuant to 28 U.S.C. §§ 1391(b), (c) and 1400(b) because Defendant is incorporated and resides in the State of Delaware and has committed acts of infringement in this District.

12. This Court has personal jurisdiction over Defendant. Defendant is a resident of this District. Defendant has and does conduct business within this District.

BACKGROUND

13. MCP incorporates the allegations of all of the foregoing paragraphs as if fully restated herein.

14. Koninklijke Philips N.V. (formerly known as Koninklijke Philips Electronics N.V.) ("Philips N.V.") and Philips North America LLC (formerly known as Philips Electronics North America Corporation) ("Philips North America") (collectively, "Philips") is a world-renowned company that engages in research and development in numerous fields. One of these fields pertains to digital video-capable devices for delivering and displaying content to users. Exemplary products in this field include desktops, laptops, all-in-one- PCs, thin clients, tablets, convertible PCs, workstations, monitors, displays, projectors, video adapters, and/or video hubs. The Asserted Patents derive from Philips's efforts in this field and claim protection for, among other things, delivering and displaying content to users.

15. Defendant made, used, sold, offered for sale, imported, tested, designed, and/or marketed in the United States digital video-capable devices for delivering and/or displaying content to users that infringe the Asserted Patents.

16. Defendant has actual notice of the Asserted Patents and of its infringement thereof. Defendant received actual notice of the Asserted Patents at least as early as March 21, 2014 by way of a letter to Defendant dated March 21, 2014. That letter included references to U.S. Patent No. 8,543,819 and U.S. Pat. App. No. 10/521,858. Defendant received a second letter dated September 16, 2020 that included allegations of infringement of the Asserted Patents. Additionally, the filing of the original Complaint and the filing of the First Amended Complaint also constitutes notice in accordance with 35 U.S.C. § 287.

17. With actual notice of the Asserted Patents, Defendant has directly infringed, and continues to directly infringe the Asserted Patents under 35 U.S.C. § 271(a) and (g) by one or more of making, using, selling and/or offering to sell, in this District and elsewhere in the United States, and importing into this District and elsewhere in the United States, certain infringing digital video-capable devices that infringe the Asserted Patents (collectively, "Accused Products"), as further described in detail in Counts I-III *infra*.

18. The Accused Products include, but are not limited to, all digital video-capable devices, including but not limited to, desktops, laptops, all-in-one- PCs, thin clients, tablets, convertible PCs, workstations, monitors, displays, projectors, video adapters, and/or video hubs, and other products that support the HDCP 2.0 protocol and above that Defendants make, use, sell, offer for sale, and/or import throughout the United States, such as: Chromebook, X360, ENVY, Elite Dragonfly, 700-Series, 800 Series, Pavilion/Pavilion Gaming, ProBook, Spectre Folio and Folio, ZBook, Create, Firefly, Studio, and OMEN laptops; Omen, Pavilion, Mini, Envy, ProDesk, and EliteDesk desktops; EliteOne and ProOne all-in-one PCs; HP 7, 8 and 10 tablets; x360 and Spectre Folio convertible PCs; Z workstations; HP Z, Omen X Emperium, DreamColor and U monitors and displays; HP HDMI 2.0 video adapters; HP Hub video adapters and HP UltraSlim, USB-C, Thunderbolt, and Z VR Backpack docks.

This list of Defendant's currently known digital video-capable devices is exemplary and, on information and belief, many other of Defendant's digital video-capable devices infringe the Asserted Patents.

19. Defendant's acts of infringement have caused damage to MCP. MCP is entitled to recover from Defendant the damages incurred by MCP as a result of Defendant's wrongful acts.

COUNT I

Defendant's Infringement of the ’809 Patent

20. MCP incorporates the allegations of all of the foregoing paragraphs as if fully restated herein.

21. Defendant has directly infringed, and continues to directly infringe, the ’809 Patent by making, using, selling, offering for sale, or importing throughout the United States products and/or methods covered by one or more claims of the ’809 Patent including, but not limited to, digital video-capable devices. The products that infringe one or more claims of the ’809 Patent include, but are not limited to, at least the Accused Products. Further discovery may reveal additional infringing products and/or models.

22. For example and without limitation, the Accused Products infringe claims 1, 17 and 49 of the ’809 Patent.

23. Attached hereto as Exhibit D, and incorporated into this Second Amended Complaint, is a claim chart showing where in the HP ProBook x360 11 G6 EE Notebook PC, Model No. 3C534UT#ABA each limitation of claims 1, 17 and 49 is met. This claim chart is exemplary and, on information and belief, many other products provided by Defendant infringe the ’809 Patent.

24. MCP is entitled to recover damages under 35 U.S.C. § 284 to adequately compensate for Defendant's infringement of the ’809 Patent.

COUNT II

Defendant's Infringement of the '186 Patent

25. MCP incorporates the allegations of all of the foregoing paragraphs as if fully restated herein.

26. Defendant has directly infringed, and continues to directly infringe, the '186 Patent by making, using, selling, offering for sale, or importing throughout the United States products and/or methods covered by one or more claims of the '186 Patent including, but not limited to, digital video-capable devices. The products that infringe one or more claims of the '186 Patent include, but are not limited to, at least the Accused Products. Further discovery may reveal additional infringing products and/or models.

27. For example and without limitation, the Accused Products infringe claim 1 of the '186 Patent.

28. Attached hereto as Exhibit E, and incorporated into this Second Amended Complaint, is a claim chart showing where in the HP ProBook x360 11 G6 EE Notebook PC, Model No. 3C534UT#ABA each limitation of claim 1 is met. This claim chart is exemplary and, on information and belief, many other products provided by Defendant infringe the '186 Patent.

29. MCP is entitled to recover damages under 35 U.S.C. § 284 to adequately compensate for Defendant's infringement of the '186 Patent.

COUNT III

Defendant's Infringement of the '564 Patent

30. MCP incorporates the allegations of all of the foregoing paragraphs as if fully restated herein.

31. Defendant has directly infringed, and continues to directly infringe, the '564 Patent by making, using, selling, offering for sale, or importing throughout the United States products and/or methods covered by one or more claims of the '564

Patent including, but not limited to, digital video-capable devices. The products that infringe one or more claims of the ’564 Patent include, but are not limited to, at least the Accused Products. Further discovery may reveal additional infringing products and/or models.

32. For example and without limitation, the Accused Products infringe claim 1 of the ’564 Patent.

33. Attached hereto as Exhibit F, and incorporated into this Second Amended Complaint, is a claim chart showing where in the HP ENVY 27 27-inch Monitor, Model No. W5A12AA#ABA each limitation of claim 1 is met. This claim chart is exemplary and, on information and belief, many other products provided by Defendant infringe the ’564 Patent.

34. MCP is entitled to recover damages under 35 U.S.C. § 284 to adequately compensate for Defendant’s infringement of the ’564 Patent.

DAMAGES

35. Defendant has refused to compensate MCP for its infringement of the Asserted Patents. MCP is entitled to monetary damages adequate to compensate MCP for Defendant’s infringement in an amount no less than a reasonable royalty for the use made of the patented inventions by Defendant. The precise amount of damages will be determined through discovery in this action and proven at trial.

MARKING

36. MCP and its licensees of the Asserted Patents have complied with 35 U.S.C. § 287, and relative to its licensees, MCP has taken reasonable steps to ensure compliance with marking.

PRAYER FOR RELIEF

WHEREFORE, MCP respectfully asks the Court for an order granting the following relief:

a) A judgment that the Asserted Patents are valid and enforceable;

b) A judgment that Defendant has directly infringed, either literally or under the Doctrine of Equivalents, one or more claims of the ’809 Patent;

c) A judgment that Defendant has directly infringed, either literally or under the Doctrine of Equivalents, one or more claims of the ’186 Patent;

d) A judgment that Defendant has directly infringed, either literally or under the Doctrine of Equivalents, one or more claims of the ’564 Patent;

e) A judgment awarding MCP all appropriate damages under 35 U.S.C. § 284 for Defendant’s past infringement, and any continuing or future infringement of the Asserted Patents, including pre and post judgment interest, costs, and disbursements pursuant to 35 U.S.C. § 284;

f) An accounting for infringing sales not presented at trial and an award by the Court of additional damages for any such infringing sales;

g) A finding that this case is exceptional within the meaning of 35 U.S.C. § 285 and that MCP be awarded its reasonable attorneys’ fees against Defendant incurred in prosecuting this action;

h) An award of reasonable attorneys’ fees, costs and expenses incurred by MCP in connection with prosecuting this action; and

i) Any and all other relief as the Court finds just, equitable, and proper under the circumstances.

DEMAND FOR JURY TRIAL

Pursuant to Fed. R. Civ. P. 38, MCP hereby respectfully demands trial by jury on all claims and issues so triable.

Dated: September 20, 2024

Respectfully submitted,

FARNAN LLP

/s/ Brian E. Farnan

Brian E. Farnan (Bar No. 4089)
Michael J. Farnan (Bar No. 5165)
919 N. Market St., 12th Floor
Wilmington, DE 19801
Phone: (302) 777-0300
Fax: (302) 777-0301
bfarnan@farnanlaw.com
mfarnan@farnanlaw.com

Michael T. Renaud (admitted *pro hac vice*)
Adam S. Rizk (admitted *pro hac vice*)
Catherine Xu (admitted *pro hac vice*)
Timothy J. Rousseau (admitted *pro hac vice*)
Courtney P. Herndon (admitted *pro hac vice*)
Williams S. Dixon (admitted *pro hac vice*)
MINTZ LEVIN COHN FERRIS GLOVSKY & POPEO PC
One Financial Center
Boston, Massachusetts 02111
Phone: (617) 542-6000
Fax: (617) 542-2241
MTRenaud@mintz.com
ARizk@mintz.com
CXu@mintz.com
TJRousseau@mintz.com
CHerndon@mintz.com
WSDixon@mintz.com

Peter F. Snell (admitted *pro hac vice*)
Brad M. Scheller (admitted *pro hac vice*)
MINTZ LEVIN COHN FERRIS GLOVSKY & POPEO PC
919 Third Avenue
New York, NY 10022

Phone: (212) 935-3000
Fax: (212) 983-3115
PFSnell@mintz.com
BMScheller@mintz.com

Attorneys for Plaintiff
Media Content Protection LLC

EXHIBIT A

US009436809B2

(12) **United States Patent**
Kamperman

(10) **Patent No.: US 9,436,809 B2**
(45) **Date of Patent: *Sep. 6, 2016**

(54) **SECURE AUTHENTICATED DISTANCE MEASUREMENT**

(71) Applicant: **KONINKLIJKE PHILIPS N.V.**, Eindhoven (NL)

(72) Inventor: **Franciscus Lucas Antonius Johannes Kamperman**, Geldrop (NL)

(73) Assignee: **KONINKLIJKE PHILIPS N.V.**, Eindhoven (NL)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **14/538,493**

(22) Filed: **Nov. 11, 2014**

(65) **Prior Publication Data**

US 2015/0074822 A1 Mar. 12, 2015

Related U.S. Application Data

(63) Continuation of application No. 10/521,858, filed as application No. PCT/IB03/02932 on Jun. 27, 2003, now Pat. No. 8,886,939.

(30) **Foreign Application Priority Data**

Jul. 26, 2002 (EP) 02078076

(51) **Int. Cl.**
G06F 21/10 (2013.01)
H04L 29/06 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC ***G06F 21/10*** (2013.01); ***H04L 63/107*** (2013.01); *G06F 2221/07* (2013.01); *G06F 2221/2111* (2013.01); *H04L 2463/101* (2013.01); *H04W 12/06* (2013.01); *H04W 24/00* (2013.01)

(58) **Field of Classification Search**
CPC G06F 21/10; H04L 63/107
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,438,824	A	3/1984	Mueller-Schloer
4,688,036	A	8/1987	Hirano et al.

(Continued)

FOREIGN PATENT DOCUMENTS

JP	9170364	A	0/6199
JP	H04306760	A	10/1992

(Continued)

OTHER PUBLICATIONS

Stefan Brands and Devid Chaum, "Distance-Bounding Protocols", Eurocrypt '93 (1993), pp. 344-359.

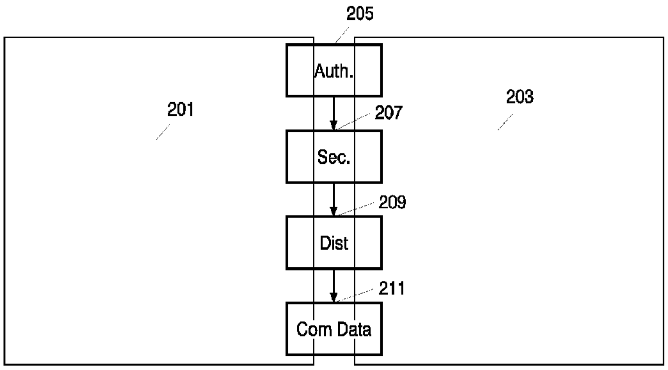
(Continued)

Primary Examiner — Darren B Schwartz

(57) **ABSTRACT**

The invention relates to a method for a first communication device to perform authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device. The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

60 Claims, 3 Drawing Sheets



(51) **Int. Cl.**

H04W 12/06	(2009.01)
H04W 24/00	(2009.01)

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,126,746	A		6/1992	Gritton	
5,596,641	A		1/1997	Ohashi et al.	
5,602,917	A		2/1997	Mueller	
5,659,617	A	*	8/1997	Fischer	H04L 9/3271 380/258
5,723,911	A		3/1998	Glehr	
5,778,071	A		7/1998	Caputo et al.	
5,937,065	A		8/1999	Simon et al.	
5,949,877	A		9/1999	Traw et al.	
5,983,347	A		11/1999	Brinkmeyer et al.	
6,085,320	A		7/2000	Kaliski, Jr.	
6,088,450	A		7/2000	Davis et al.	
6,151,676	A		11/2000	Cuccia et al.	
6,208,239	B1		3/2001	Muller et al.	
6,346,878	B1		2/2002	Pohlman et al.	
6,351,235	B1		2/2002	Stilp	
6,442,690	B1	*	8/2002	Howard, Jr.	G06F 21/602 713/156
6,484,948	B1		11/2002	Sonoda	
6,493,825	B1		12/2002	Blumenau et al.	
6,526,509	B1	*	2/2003	Horn	H04L 9/3263 380/277
6,550,011	B1	*	4/2003	Sims, III	G06F 21/10 365/52
7,200,233	B1		4/2007	Keller et al.	
8,107,627	B2		1/2012	Epstein	
8,352,582	B2		1/2013	Epstein	
8,997,243	B2		3/2015	Epstein	
2001/0008558	A1		7/2001	Hirafuji	
2001/0043702	A1		11/2001	Elteto et al.	
2001/0044786	A1		11/2001	Ishibashi	
2001/0050990	A1	*	12/2001	Sudia	G06Q 20/02 380/286
2002/0007452	A1		1/2002	Traw et al.	
2002/0026424	A1		2/2002	Akashi	
2002/0026576	A1		2/2002	Das-Purkayastha et al.	
2002/0035690	A1		3/2002	Nakano	
2002/0061748	A1		5/2002	Nakakita et al.	
2002/0078227	A1		6/2002	Kronenberg	
2002/0166047	A1	*	11/2002	Kawamoto	H04L 9/3263 713/169
2003/0021418	A1		1/2003	Arakawa et al.	
2003/0030542	A1		2/2003	von Hoffmann	
2003/0051151	A1	*	3/2003	Asano	G11B 20/00086 713/193
2003/0065918	A1		4/2003	Willey	
2003/0070092	A1		4/2003	Hawkes et al.	
2003/0112978	A1		6/2003	Rodman et al.	
2003/0184431	A1		10/2003	Lundkvist	
2003/0220765	A1		11/2003	Overy et al.	
2004/0015693	A1		1/2004	Kitazumi	
2004/0080426	A1	*	4/2004	Fraenkel	H04W 8/245 340/9.14
2005/0114647	A1		5/2005	Epstein	
2005/0265503	A1		12/2005	Rofheart et al.	
2006/0294362	A1		12/2006	Epstein	

FOREIGN PATENT DOCUMENTS

JP	H0619948	A	1/1994
JP	H08234658	A	9/1996
JP	H09170364	A	6/1997
JP	11101035	A	4/1999
JP	11208419	A	8/1999
JP	2000357156	A	12/2000
JP	2001249899	A	9/2001
JP	2001257672	A	9/2001
JP	2002124960	A	4/2002
JP	2002189966	A	7/2002
WO	9739553	A1	10/1997
WO	9949378		9/1999
WO	0152234	A1	7/2001
WO	0193434	A2	12/2001
WO	0233887	A2	4/2002
WO	0235036	A1	5/2002

OTHER PUBLICATIONS

Tim Kindber & Kan Zhang, "Context Authentication Using Constrained Channels", pp. 1-8.

Hitachi, Ltd, "5C Digital Transmission Content Protection White Papter", Rev. 1.0, July 14, 1998, pp. 1013.

Boyd et al, "Protocols for Authentication and Key Establishment", Spring-Verlag, September 17, 2003, pp. 116-120, 195-195, 305.

Modern Cryptography Theory (1986) Chapter 9, ISBN: 4-88552-064-9 (Japanese).

Hayashi et al, Encyption and Authentication Program Module, Technical Paper (Japanese) NTT R&D vol. 44 No. 10 Oct. 1, 1995.

Ikeno et al. "Modern Cryptography Theory" Japan, Institute of Electronics, Information and Communication Engineersm Nov. 15, 1997, p. 175-177.

* cited by examiner

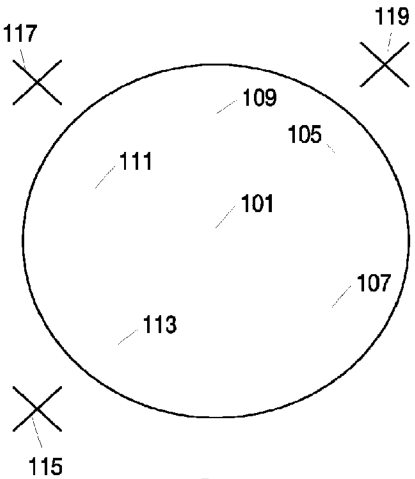

117
119
109
105
111
101
107
113
115

FIG. 1

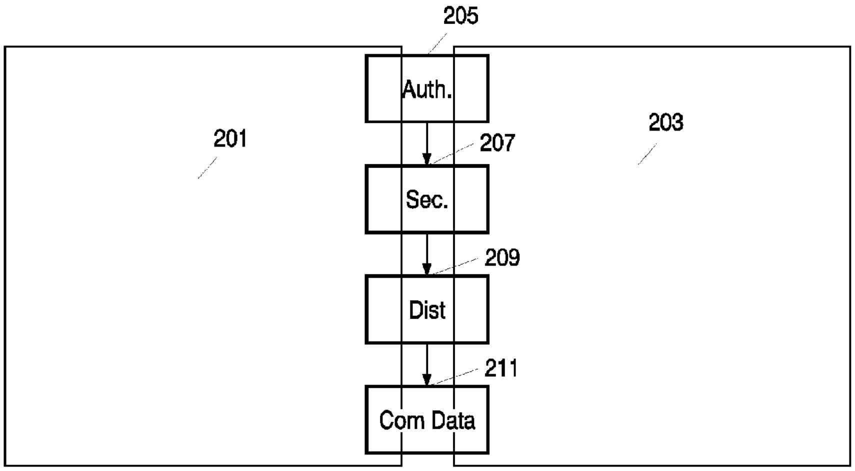

205
Auth.
201
207
203
Sec.
209
Dist
211
Com Data

FIG. 2

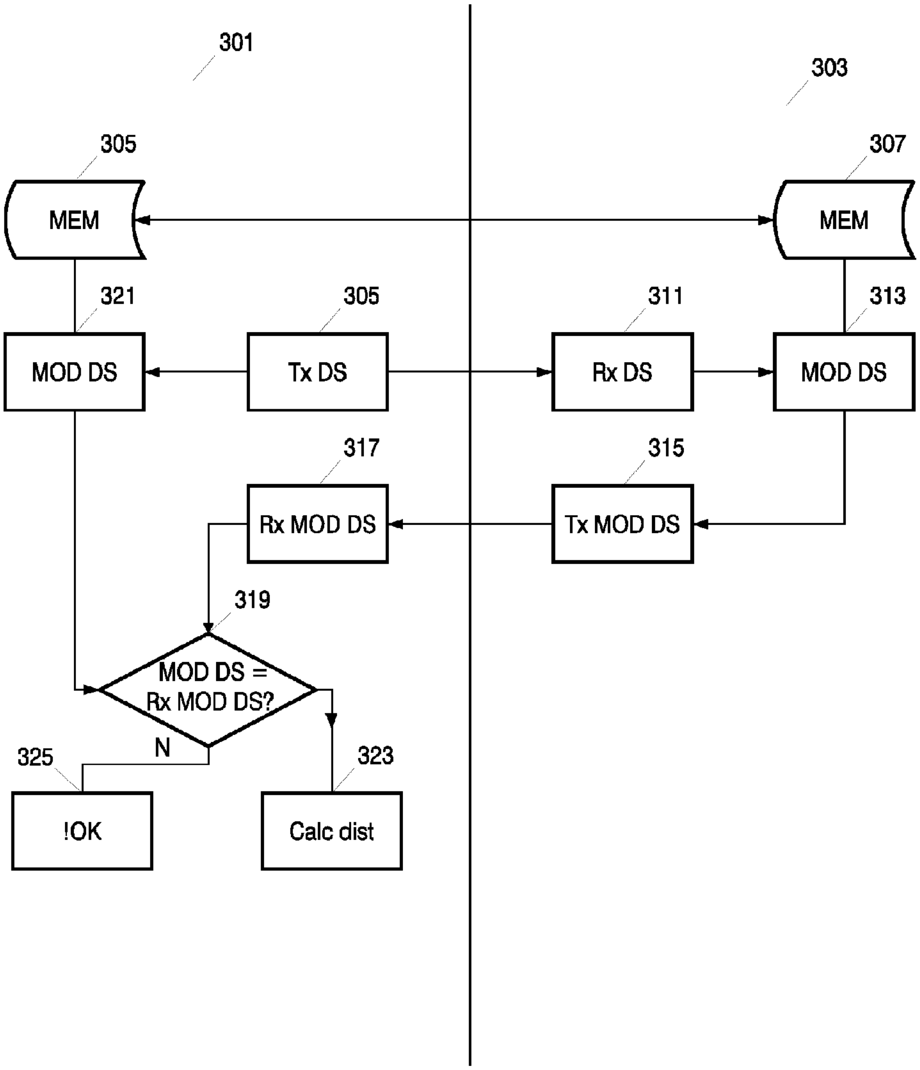

301
303
305
MEM
307
MEM
321
MOD DS
305
Tx DS
311
Rx DS
313
MOD DS
317
Rx MOD DS
315
Tx MOD DS
319
MOD DS =
Rx MOD DS?
N
325
!OK
323
Calc dist

FIG. 3

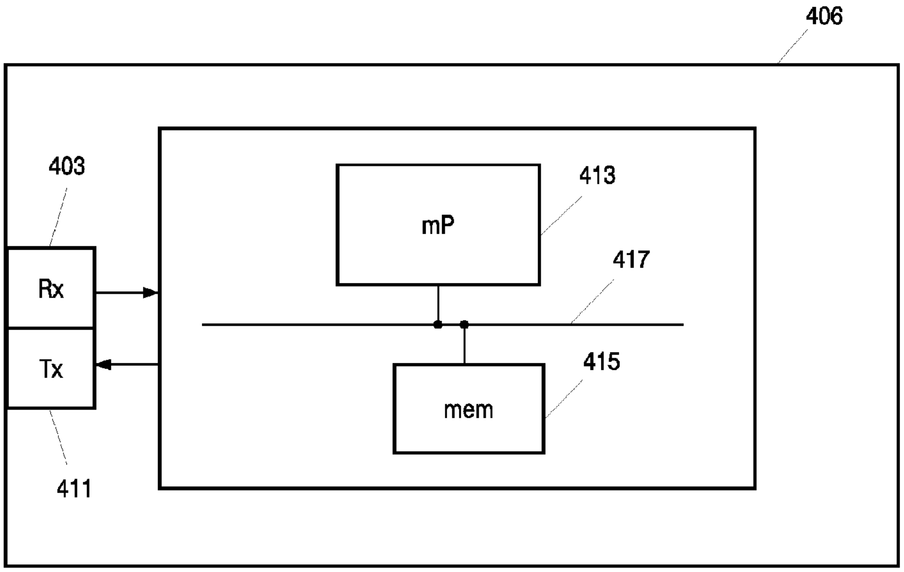


FIG. 4

SECURE AUTHENTICATED DISTANCE MEASUREMENT

This application claims, pursuant to 35 USC 120, priority to and the benefit of the earlier filing date of, that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), which claimed priority to and the benefit of the earlier filing date, as a National Stage Filing of that international patent application filed on Jun. 27, 2003 and afforded serial number PCT/IB03/02932 (WO2004014037), which claimed priority to and the benefit of the earlier filing date of that patent application filed on Jul. 26, 2002 and afforded serial number EP02078076.3, the contents of all of which are incorporated by reference, herein.

This application is further related to that patent application entitled "Secure authenticated Distance Measurement", filed on Jul. 24, 2009 and afforded Ser. No. 12/508,917 (now U.S. Pat. No. 8,543,819), issued Sep. 24, 2013), which claimed priority to and the benefit of the earlier filing date of that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), the contents of which are incorporated by reference herein.

The invention relates to a method for a first communication device to perform authenticated distance measurement between a first communication device and a second communication device. The invention also relates to a method of determining whether data stored on a first communication device is to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

Digital media have become popular carriers for various types of data information. Computer software and audio information, for instance, are widely available on optical compact disks (CDs) and recently also on digital video/versatile discs (DVDs) which have been gaining in distribution share. The CD and the DVD utilize a common standard for the digital recording of data, software, images, audio and multimedia. Additional media, such as recordable discs, solid-state memory, and the like, are making considerable gains in the software and data distribution market.

The substantially superior quality of the digital format as compared to the analog format renders the former substantially more prone to unauthorized copying and pirating, further a digital format is both easier and faster to copy. Copying of a digital data stream, whether compressed, uncompressed, encrypted or non-encrypted, typically does not lead to any appreciable loss of quality in the data. Digital copying thus is essentially unlimited in terms of multi-generation copying. Analog data with its signal to noise ratio loss with every sequential copy, on the other hand, is naturally limited in terms of multi-generation and mass copying.

The advent of the recent popularity in the digital format has also brought about a slew of copy protection and digital rights management (DRM) systems and methods. These systems and methods use technologies such as encryption, watermarking and right descriptions (e.g. rules for accessing and copying data).

One way of protecting content in the form of digital data is to ensure that content will only be transferred between devices if:

- the receiving device has been authenticated as being a compliant device, and
- the user of the content has the right to transfer (move, copy) that content to another device.

If transfer of content is allowed, this will typically be performed in an encrypted way to make sure that the content cannot be captured illegally in a useful format.

Technology to perform device authentication and encrypted content transfer is available and is called a secure authenticated channel (SAC). Although it might be allowed to make copies of content over a SAC, the content industry is very bullish on content distribution over the Internet. This results in disagreement of the content industry on transferring content over interfaces that match well with the Internet, e.g. Ethernet.

Further, it should be possible for a user visiting his neighbor to watch a movie, which he owns, on the neighbor's big television screen. Typically, the content owner will disallow this, but it might become acceptable if it can be proved that a license holder of that movie (or a device that the license holder owns) is near that television screen.

It is therefore of interest to be able to include an authenticated distance measurement when deciding whether content should be accessed or copied by other devices.

In the article by Stefan Brands and David Chaum, "Distance-Bounding protocols", Eurocrypt '93 (1993), Pages 344-359, integration of distance-bounding protocols with public-key identification schemes is described. Here distance measurement is described based on time measurement using challenge and response bits and with the use of a commitment protocol. This does not allow authenticated device compliancy testing and is not efficient when two devices must also authenticate each other.

It is an object of the invention to obtain a solution to the problem of performing a secure transfer of content within a limited distance.

This is obtained by a method for a first communication device to performing authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device.

Because the common secret is being used for performing the distance measurement, it can be ensured that when measuring the distance from the first communication device to the second communication device, it is the distance between the right devices that is being measured.

The method combines a distance measurement protocol with an authentication protocol. This enables authenticated device compliancy testing and is efficient, because a secure channel is anyhow needed to enable secure communication between devices and a device can first be tested on compliancy before a distance measurement is executed.

In a specific embodiment, the authenticated distance measurement is performed according to the following steps;

- transmitting a first signal from the first communication device to the second communication device at a first time t**1**, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal to the first device,
- receiving the second signal at a second time t**2**,
- checking if the second signal has been modified according to the common secret, and

- determining the distance between the first and the second communication device according to a time difference between t**1** and t**2**.

When measuring a distance by measuring the time difference between transmitting and receiving a signal and using a secret, shared between the first and the second communication device, for determining whether the returned signal really originated from the second communication device, the distance is measured in a secure authenticated way ensuring that the distance will not be measured to a third communication device (not knowing the secret). Using the shared secret for modifying the signal is a simple way to perform a secure authenticated distance measurement.

In a specific embodiment, the first signal is a spread spectrum signal. Thereby a high resolution is obtained and it is possible to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In another embodiment the step of checking if the second signal has been modified according to the common secret is performed by the steps of:

- generating a third signal by modifying the first signal according to the common secret, and
- comparing the third signal with the received second signal.

This method is an easy and simple way of performing the check, but it requires that both the first communication device and the second communication device know how the first signal is being modified using the common secret.

In a specific embodiment the first signal and the common secret are bit words and the second signal comprises information being generated by performing an exclusive OR operation (XOR) between the bit words. Thereby, it is a very simple operation that has to be performed, resulting in demand for few resources by both the first and the second communication device when performing the operation.

In an embodiment, the common secret has been shared before performing the distance measurement, the sharing being performed by the steps of:

- performing an authentication check from the first communication device on the second communication device by checking whether the second communication device is compliant with a set of predefined compliance rules, and
- if the second communication device is compliant, sharing the common secret by transmitting the secret to the second communication device.

This is a secure way of performing the sharing of the secret, ensuring that only devices being compliant with compliance rules can receive the secret. Further, the shared secret can afterwards be used for generating a SAC channel between the two devices. The secret could be shared using e.g. key transport mechanisms as described in ISO 11770-3. Alternatively, a key agreement protocol could be used, which e.g. is also described in ISO 11770-3.

In another embodiment the authentication check further comprises checking if the identification of the second device is compliant with an expected identification. Thereby, it is ensured that the second device really is the device that it should be. The identity could be obtained by checking a certificate stored in the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between the first and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. By using the authenticated distance measurement in connection with sharing data between devices, unauthorized distribution of content can be reduced.

In a specific embodiment the data stored on the first device is sent to the second device if it is determined that the data stored on the first device are to be accessed by the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between a third communication device and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. In this embodiment, the distance is not measured between the first communication device, on which the data are stored, and the second communication device. Instead, the distance is measured between a third communication device and the second communication device, where the third communication device could be personal to the owner of the content.

The invention also relates to a communication device for performing authenticated distance measurement to a second communication device, where the communication device shares a common secret with the second communication device and where the communication device comprises means for measuring the distance to the second device using the common secret.

In an embodiment, the device comprises:

- means for transmitting a first signal to a second communication device at a first time t**1**, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal,
- means for receiving the second signal at a second time t**2**,
- means for checking if the second signal has been modified according to the common secret, and
- means for determining the distance between the first and the second communication device according to a time difference between t**1** and t**2**.

The invention also relates to an apparatus for playing back multimedia content comprising a communication device according to the above.

In the following preferred embodiments of the invention will be described referring to the figures, wherein:

FIG. **1** illustrates authenticated distance measurement being used for content protection,

FIG. **2** is a flow diagram illustrating the method of performing authenticated distance measurement,

FIG. **3** illustrates in further detail the step of performing the authenticated distance measurement shown in FIG. **2**, and

FIG. **4** illustrates a communication device for performing authenticated distance measurement.

FIG. **1** illustrates an embodiment wherein the authenticated distance measurement is being used for content protection. In the center of the circle **101** a computer **103** is placed. The computer comprises content, such as data, software, images, multimedia content being video and/or audio, stored on e.g. a hard disk, solid state memory, a DVD or a CD. The owner of the computer **103** owns the content and therefore the computer is authorized to access and present the multimedia content for the user. When the user wants to make a legal copy of the content on another device via e.g. a SAC, the distance between the other device and the computer **103** is measured and only devices within a predefined distance illustrated by the devices **105**, **107**, **109**, **111**, **113** inside the circle **101** are allowed to receive the content. Whereas the devices **115**, **117**, **119** having a distance to the computer **103** being larger than the predefined distance are not allowed to receive the content.

In the example a device is a computer **103**, but it could e.g. also be a DVD drive, a CD drive or a Video display device, as long as the device comprises a communication device for performing the distance measurement.

In a specific example, the distance might not be measured between the computer **103**, on which the data are stored, and the other device, it could be determined between a third device (e.g. a device being personal to the owner of the content and which does not contain the data) and the other device.

In FIG. **2** a flow diagram illustrates the general idea of performing authenticated distance measurement between two devices, **201** and **203** each comprising communication devices for performing the authenticated distance measurement. In the example the first device **201** comprises content which the second device **203** has requested. The authenticated distance measurement then is as follows. In step **205** the first device **201** authenticates the second device **203**; this could comprise the steps of checking whether the second device **203** is a compliant device and might also comprise the step of checking whether the second device **203** really is the device identified to the first device **201**. Then in step **207**, the first device **201** exchanges a secret with the second device **203**, which e.g. could be performed by transmitting a random generated bit word to the second device **203**. The secret should be shared securely, e.g. according to some key management protocol as described in e.g. ISO 11770.

Then in step **209**, a signal for distance measurement is transmitted to the second device **203**; the second device modifies the received signal according to the secret and retransmits the modified signal back to the first device. The first device **201** measures the round trip time between the signal leaving and the signal returning and checks if the returned signal was modified according to the exchanged secret. The modification of the returned signal according to some secret will most likely be dependent on the transmission system and the signal used for distance measurement, i.e. it will be specific for each communication system (such as 1394, Ethernet, Bluetooth, IEEE 802.11, etc.).

The signal used for the distance measurement may be a normal data bit signal, but also special signals other than for data communication may be used. In an embodiment spread spectrum signals are used to be able to get high resolution and to be able to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In a specific example a direct sequence spread spectrum signal is used for distance measurement; this signal could be modified by XORing the chips (e.g. spreading code consisting of 127 chips) of the direct sequence code by the bits of the secret (e.g. secret consists also of 127 bits). Also, other mathematical operations similar to XOR could be used.

The authentication **205** and exchange of secret **207** could be performed using the protocols described in some known ISO standards e.g. ISO 9798 and ISO 11770. For example the first device **201** could authenticate the second device **203** according to the following communication scenario:

First device->Second device: $R_B \| \text{Text } 1$
where R_B is a random number
Second device->First device: CertA||TokenAB
Where CertA is a certificate of A
$\text{TokenAB}=R_A \| R_B \| B \| \text{Text3} \| sS_A(R_A \| R_B \| B \| \text{Text2})$
R_A is a random number
Indentifier B is an option
sS_A is a signature set by A using private key S_A

If TokenAB is replaced with the token as specified in ISO 11770-3 we at the same time can do secret key exchange. We can use this by substituting Text2 by:

$\text{Text2}:=eP_B(A \| K \| \text{Text2}) \| \text{Text3}$
Where eP_B is encrypted with Public key B
A is identifier of A
K is a secret to be exchanged

In this case the second device **203** determines the key (i.e. has key control), this is also called a key transport protocol, but also a key agreement protocol could be used. This may be undesirable in which case it can be reversed, such that the first device determines the key. A secret key has now been exchanged according to step **207** in FIG. **2**. Again, the secret key could be exchanged by e.g. a key transport protocol or a key agreement protocol.

After the distance has been measured in a secure authenticated way as described above, content data can be sent between the first and the second device in step **211** in FIG. **2**.

FIG. **3** illustrates in further detail, the step of performing the authenticated distance measurement. As described above, the first device **301** and the second device **303** have exchanged a secret; the secret is stored in the memory **305** of the first device and the memory **307** of the second device. In order to perform the distance measurement, a signal is transmitted to the second device via a transmitter **305**. The second device receives the signal via a receiver **311**, and microprocessor **313** modifies the signal by using the locally stored secret. The signal is modified by the second device according to rules known by the first device **301** and transmitted back to the first device **301** via a transmitter **315**. The first device **301** receives the modified signal via a receiver **317** and in **319** the received modified signal is compared to a signal, which has been modified locally i.e. by the first device. The local modification is performed in microprocessor **321** by using the signal transmitted to the second device in transmitter **305** and then modifying the signal using the locally stored secret similar to the modification rules used by the second device. If the received modified signal and the locally modified signal are identical, then the received signal is authenticated and can be used for determining the distance between the first and the second device. If the two signals are not identical, then the received signal cannot be authenticated and can therefore not be used for measuring the distance as illustrated by **325**. In microprocessor **323** the distance is calculated between the first and the second device; this could e.g. be performed by measuring the time, when the signal is transmitted by the transmitter **309** from the first device to the second device and measuring when the receiver **317** receives the signal from the second device. The time difference between a transmittal time and a reception time can then be used for determining the physical distance between the first device and the second device.

In FIG. **4** a communication device for performing authenticated distance measurement is illustrated. The device **406** comprises a receiver **403** and a transmitter **411**. The device further comprises means for performing the steps described above, which could be performed by executing software using a microprocessor **413** connected to memory **415** via a communication bus **417**. The communication device could then be placed inside devices such as a DVD, a DVD recorder, a computer, a CD, a CD recorder, a solid state memory, a television and other devices for providing protected content, accessing protected content, or authorizing the access to protected content.

What is claimed is:

1. A first device for controlling delivery of protected content to a second device, the first device comprising:

a memory;

a processor, said processor arranged to:

receive a certificate of the second device, the certificate providing information regarding the second device;

determine whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate;

provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;

receive a second signal from the second device after providing the first signal;

determine whether the second signal is derived from a secret known by the first device;

determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and

allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.

2. The first device of claim **1**, wherein the first signal comprises a random number.

3. The first device of claim **1**, wherein the second signal is formed by modifying the first signal based on the secret, wherein the modification comprises performing an XOR operation on the first signal.

4. The first device of claim **1**, wherein the processor is further arranged to provide the secret to the second device.

5. The first device of claim **4**, wherein the secret is securely provided using one of: a key transport protocol, a key management protocol and a key agreement protocol.

6. The first device of claim **4**, wherein the processor arranged to provide the secret to the second device comprises the processor arranged to provide the secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number.

7. The first device of claim **1**, wherein the processor is further arranged to receive the secret from the second device.

8. The first device of claim **7**, wherein the secret is securely received using one of: a key transport protocol, a key management protocol and a key agreement protocol.

9. The first device of claim **1**, wherein the processor arranged to determine whether the second signal is derived from the secret is arranged to:

modify the first signal according to the secret;

compare the modified first signal with the second signal; and

provide an indication when said modified first signal is identical to the second signal.

10. The first device of claim **1**, wherein the first signal and the secret are of comparable length.

11. The first device of claim **1**, wherein the processor is further arranged to determine an identity of the second device using the certificate.

12. The first device of claim **1**, wherein the certificate comprises a public key.

13. The first device of claim **1**, wherein the processor is further arranged to provide a certificate to the second device.

14. The first device of claim **1**, wherein the predetermined time is based on a communication system associated with the first device.

15. The first device of claim **1**, wherein the second signal comprises the first signal modified by the secret.

16. The first device of claim **1**, wherein the processor is further arranged to:

provide instruction to a third device to transmit said protected content, wherein said protected content is stored on said third device.

17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:

means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;

means for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate;

means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules;

means for receiving a second signal at a second time from the requesting device;

means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and

a time difference between the first time and the second time is less than a predetermined time.

18. The system of claim **17**, wherein said protected content is stored on a third device.

19. The system of claim **18**, wherein said means for providing the requested content comprises:

means for providing instruction to said third device to provide said content to said requesting device.

20. The system of claim **18**, wherein the third device is one of: a DVD, CD and a storage device.

21. The system of claim **17**, wherein the secret is securely received by the content provider.

22. The system of claim **17**, wherein the secret is securely transmitted by the content provider.

23. The system of claim **17**, wherein the certificate identifies the requesting device.

24. The system of claim **17**, wherein the predetermined time is based on a type of communication protocol between the requesting device and the content provider.

25. The system of claim **17**, wherein the content provider is one of: a DVD, CD and a storage device.

26. The system of claim **17**, wherein the second signal comprises the first signal modified by the secret.

27. A first device in communication with a second device, the first device comprising:

a memory;

a processor in communication with the memory, the processor arranged to execute software stored on the first device, the software configured to:

receive from the second device a request for a protected content and a certificate providing information associated with the second device;

determine whether the second device is suitable for receiving said protected content, wherein determining suitability of said second device is based on said information provided in said certificate;

provide a first signal to said second device when said second device is determined to be suitable for receiving said protected content;

receive from said second device a second signal;

determine whether said second signal is representative of said first signal modified according to a secret known by said first device and said second device;

determine whether a time difference between a time of providing the first signal and receiving the second signal is less than a predetermined time; and

initiate transmission of said protected content to said second device when at least said second signal is representative of said first signal modified according to a secret known by said first device and said second device and said time difference is less than the predetermined time.

28. The first device of claim **27**, wherein said protected content is stored on said first device.

29. The first device of claim **27**, wherein the software configured to initiate said initiating transmission of said protected content is further configured to provide instruction to a third device to transmit said protected content, wherein said protected content is stored on said third device.

30. The first device of claim **29**, wherein said third device is one of a DVD, a CD and a storage device.

31. The first device of claim **29**, wherein said third device is remotely located from said first device.

32. The first device of claim **27**, wherein suitability is determined as being compliant with a set of compliancy rules.

33. The first device of claim **27**, wherein the software is further arranged to:

provide the secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is suitable, said secret comprising a random number.

34. A method of a first device controlling delivery of protected content to a second device, the method comprising:

receiving a certificate of the second device, the certificate providing information regarding the second device;

determining whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate;

providing a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;

receiving a second signal from the second device after providing the first signal;

determining whether the second signal is derived from a secret known by the first device;

determining whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and

allowing the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.

35. The method of claim **34**, wherein the first signal comprises a random number.

36. The method of claim **34**, wherein the second signal is formed by modifying the first signal based on the secret, wherein the modification comprises performing an XOR operation on the first signal.

37. The method of claim **34**, further comprising providing the secret to the second device.

38. The method of claim **37**, wherein the secret is securely provided using one of: a key transport protocol, a key management protocol and a key agreement protocol.

39. The method of claim **34**, further comprising receiving the secret from the second device.

40. The method of claim **39**, wherein the secret is securely received using one of: a key transport protocol, a key management protocol and a key agreement protocol.

41. The method of claim **34**, wherein the step of determining whether the second signal is derived from the secret comprises:

modifying the first signal according to the secret;

comparing the modified first signal with the second signal; and

providing an indication when said modified first signal is identical to the second signal.

42. The method of claim **34**, wherein the first signal and the secret are of comparable length.

43. The method of claim **34**, further comprising determining an identity of the second device using the certificate.

44. The method of claim **34**, wherein the certificate comprises a public key.

45. The method of claim **34**, further comprising providing a certificate to the second device.

46. The method of claim **34**, wherein the predetermined time is based on a communication system associated with the first device.

47. The method of claim **34**, wherein the second signal comprises the first signal modified by the secret.

48. The method of claim **34**, further comprising providing instruction to a third device to transmit said protected content, wherein said protected content is stored on said third device.

49. A first device for controlling delivery of protected content to a second device, the first device comprising:

a memory;

a processor, the processor arranged to:

receive a certificate from the second device prior to sending a first signal;

determine from the certificate if the second device is compliant;

provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;

provide the first signal to the second device;

receive a second signal from the second device after providing the first signal;

determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;

determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and

allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.

50. The first device of claim **49**, wherein the processor is further arranged to:

use the secret to generate a secure authenticated channel between the first device and the second device,

use the secure authenticated channel to provide the protected content to the second device.

51. The first device of claim **49**, wherein the secret and the first signal are of comparable length.

52. The first device of claim **49**, wherein the modification is a XOR operation using the first signal.

53. The first device of claim **49**, wherein the processor, arranged to determine that the second signal is derived from the secret, is further arranged to:

modify the first signal according to the secret;

compare the modified first signal with the second signal; and

determine that the modified first signal is identical to the second signal.

54. The first device of claim **49**, wherein the first signal comprises a random number.

55. A method of a first device controlling delivery of protected content to a second device, the method comprising:

receiving a certificate from the second device prior to sending a first signal;

determining from the certificate if the second device is compliant;

providing a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;

providing the first signal to the second device;

receiving a second signal from the second device after providing the first signal;

determining if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;

determining whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and

allowing the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.

56. The method of claim **55**, further comprising:

using the secret to generate a secure authenticated channel between the first device and the second device,

using the secure authenticated channel to provide the protected content to the second device.

57. The method of claim **55**, wherein the secret and the first signal have the same bit length.

58. The method of claim **55**, wherein the modification is a XOR operation using the first signal.

59. The method of claim **55**, wherein the step of determining that the second signal is derived from the secret comprises:

modifying the first signal according to the secret;

comparing the modified first signal with the second signal; and

determining that the modified first signal is identical to the second signal.

60. The method of claim **55**, wherein the first signal comprises a random number.

* * * * *

EXHIBIT B

US010091186B2

(12) **United States Patent**
Kamperman

(10) **Patent No.: US 10,091,186 B2**
(45) **Date of Patent: *Oct. 2, 2018**

(54) **SECURE AUTHENTICATED DISTANCE MEASUREMENT**

(71) Applicant: **KONINKLIJKE PHILIPS N.V.**, Eindhoven (NL)

(72) Inventor: **Franciscus L. A. J. Kamperman**, Geldrop (NL)

(73) Assignee: **Koninklijke Philips N.V.**, Eindhoven (NL)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 72 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **15/352,646**

(22) Filed: **Nov. 16, 2016**

(65) **Prior Publication Data**

US 2017/0063556 A1 Mar. 2, 2017

Related U.S. Application Data

(63) Continuation of application No. 15/229,207, filed on Aug. 5, 2016, now Pat. No. 9,590,977, which is a (Continued)

(30) **Foreign Application Priority Data**

Jul. 26, 2002 (EP) 02078076

(51) **Int. Cl.**

H04L 29/06	(2006.01)
H04L 9/14	(2006.01)

(Continued)

(52) **U.S. Cl.**
CPC ***H04L 63/0823*** (2013.01); ***G06F 21/10*** (2013.01); ***H04L 9/14*** (2013.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 63/0823; H04L 9/14; H04L 9/30; H04L 9/3263; H04L 63/062; H04L 43/16;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,438,824	A	3/1984	Mueller-Scholoer
4,688,036	A	8/1987	Hirano et al.

(Continued)

FOREIGN PATENT DOCUMENTS

EP	1100035	A1	5/2001
JP	H04306760	A	10/1992

(Continued)

OTHER PUBLICATIONS

Ikeno et al "Modern Cryptography Theory" Japan, Institute of Electronics, Information and Communication Engineers, Nov. 15, 1997, p. 175-177.

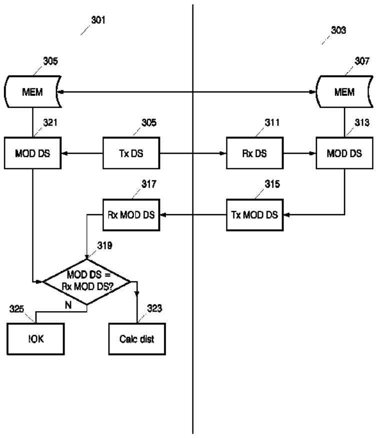
(Continued)

Primary Examiner — Darren B Schwartz

(57) **ABSTRACT**

The invention relates to a method for a first communication device to perform authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device. The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

36 Claims, 3 Drawing Sheets



Related U.S. Application Data

continuation of application No. 14/538,493, filed on Nov. 11, 2014, now Pat. No. 9,436,809, which is a continuation of application No. 10/521,858, filed as application No. PCT/IB03/02932 on Jun. 27, 2003, now Pat. No. 8,886,939.

(51) **Int. Cl.**

H04L 9/30	(2006.01)
H04L 9/32	(2006.01)
H04L 12/26	(2006.01)
G06F 21/10	(2013.01)
H04W 24/00	(2009.01)
H04W 12/06	(2009.01)

(52) **U.S. Cl.**

CPC ***H04L 9/30*** (2013.01); ***H04L 9/3263*** (2013.01); ***H04L 43/0852*** (2013.01); ***H04L 43/16*** (2013.01); ***H04L 63/062*** (2013.01); ***H04L 63/107*** (2013.01); *G06F 2221/07* (2013.01); *G06F 2221/2111* (2013.01); *H04L 63/0428* (2013.01); *H04L 2463/101* (2013.01); *H04W 12/06* (2013.01); *H04W 24/00* (2013.01)

(58) **Field of Classification Search**

CPC H04L 43/0852; H04L 63/107; H04L 63/0428; H04L 2463/101; G06F 21/10; G06F 2221/07; G06F 2221/2111; H04W 24/00; H04W 12/06

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,926,480	A		5/1990	Chaum	
5,126,746	A		6/1992	Gritton	
5,351,293	A	*	9/1994	Michener	H04L 9/0822 380/44
5,596,641	A		1/1997	Ohashi et al.	
5,602,917	A		2/1997	Mueller	
5,659,617	A		8/1997	Fischer	
5,708,712	A	*	1/1998	Brinkmeyer	B60R 25/04 340/5.25
5,723,911	A		3/1998	Glehr	
5,778,071	A		7/1998	Caputo et al.	
5,937,065	A		8/1999	Simon et al.	
5,949,877	A	*	9/1999	Traw	G06F 21/10 380/30
5,983,347	A		11/1999	Brinkmeyer et al.	
6,085,320	A		7/2000	Kaliski	
6,088,450	A		7/2000	Davis et al.	
6,148,404	A	*	11/2000	Yatsukawa	G06F 21/335 380/30
6,151,676	A		11/2000	Cuccia et al.	
6,208,239	B1		3/2001	Muller et al.	
6,346,878	B1		2/2002	Pohlman et al.	
6,351,235	B1		2/2002	Stilp	
6,442,690	B1		8/2002	Howard, Jr.	
6,484,948	B1		11/2002	Sonoda	
6,493,825	B1		12/2002	Blumenau et al.	
6,526,598	B1		3/2003	Horn	
6,550,011	B1		4/2003	Sims	
7,200,233	B1		4/2007	Keller et al.	
7,242,766	B1		7/2007	Lyle	
7,516,325	B2		4/2009	Willey	
7,685,423	B1	*	3/2010	Walmsley	G06F 21/44 399/24
7,787,865	B2		8/2010	Willey	
7,898,977	B2		3/2011	Roese	
8,068,610	B2		11/2011	Moroney	
8,107,627	B2		1/2012	Epstein	
8,352,582	B2		1/2013	Epstein	
8,997,243	B2		3/2015	Epstein	
2001/0002486	A1	*	5/2001	Kocher	G06F 7/723 713/171
2001/0008558	A1		7/2001	Hirafuji	
2001/0043702	A1		11/2001	Elteto et al.	
2001/0044786	A1		11/2001	Ishibashi	
2001/0050990	A1	*	12/2001	Sudia	G06Q 20/02 380/286
2002/0007452	A1	*	1/2002	Traw	G06F 21/10 713/152
2002/0026424	A1		2/2002	Akashi	
2002/0026576	A1		2/2002	Das-Purkayastha et al.	
2002/0035690	A1		3/2002	Nakano	
2002/0061748	A1		5/2002	Nakakita et al.	
2002/0078227	A1		6/2002	Kronenberg	
2002/0166047	A1		11/2002	Kawamoto	
2003/0021418	A1		1/2003	Arakawa et al.	
2003/0030542	A1		2/2003	Von Hoffmann	
2003/0051151	A1		3/2003	Asano	
2003/0065918	A1		4/2003	Willey	
2003/0070092	A1		4/2003	Hawkes et al.	
2003/0112978	A1		6/2003	Rodman et al.	
2003/0184431	A1		10/2003	Lundkvist	
2003/0220765	A1		11/2003	Overy et al.	
2004/0015693	A1		1/2004	Kitazumi	
2004/0080426	A1		4/2004	Fraenkel	
2005/0114647	A1		5/2005	Epstein	
2005/0265503	A1		12/2005	Rofheart et al.	
2006/0294362	A1		12/2006	Epstein	

FOREIGN PATENT DOCUMENTS

JP	H0619948	A	1/1994
JP	H08234658	A	9/1996
JP	9170364	A	6/1997
JP	H09170364	A	6/1997
JP	11101035	A	4/1999
JP	11208419	A	8/1999
JP	2000357156	A	12/2000
JP	2001249899	A	9/2001
JP	2001257672	A	9/2001
JP	2002124960		4/2002
JP	2002189966	A	7/2002
WO	9739553	A1	10/1997
WO	9949378		9/1999
WO	0152234	A1	7/2001
WO	0193434	A1	12/2001
WO	0233887	A2	4/2002
WO	0235036	A1	5/2002
WO	02054353	A1	7/2002

OTHER PUBLICATIONS

Modern Cryptography Theory (1986) Chapter 9, ISBN: 4-88552-064-9 (Japanese).

Hayashi et al Encryption and Authentication Program Module , Technical Paper (Japanese) NTT R&D vol. 44, No. 10 Oct. 1, 1995.

Stefan Brands and Devid Chaum "Distance Bounding Protocols" Eurocrypt "93, (1993) p. 344-359.

Tim Kindber & Kan Zhang "Context Authention Using Constrained Channels" pp. 1-8 , Apr. 16, 2001.

Hitachi Ltd., 5C Digital Transmission Content Protection White Paper Rev. 1.0 Jul. 14, 1998, p. 1013.

Boyd et al "Protocols for Authention and Key Establishment" Spring-Verlag, Sep. 17, 2003, p. 116-120, 195, 305.

SmartRight™ Certification for FCC Approval for Use with the Broadcast Flag, Mar. 1, 2004.

SmartRight™ Copy Protection for System for Digital Home Networks, Deployment Process, CPTWG, Nov. 28, 2001.

SmartRight™ Copy Protection System for Digital Home Networks, CPTWG, May 24, 2001.

SmartRight™ Digital Broadcast Content Protection, Presentation to FCC, Apr. 2, 2004 (cited in litigation).

SmartRight™ Technical White Paper, Version 1.7, Jan. 2003 ("White Paper") (cited in litigation).

(56) References Cited

OTHER PUBLICATIONS

Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2408 (“RFC 2408”)—cited in litigation.
International Standard ISO/IEC 11770-3 (1st ed.) (“ISO 11770-3”) , 2008.
Scott Crosby, et al., “A Cryptanalysis of the High-bandwidth Digital Content Protection System” Computer and Communications Security, (2001).
SmartRight™ Specifications Sep. 26, 2001.
SmartRight™ Copy Protection System for Digital Home Networks, CPTWG, Jul. 11, 2001.
Bruce Schneier, Applied Cryptography (2d ed. 1996) (“Schneier”).
Steven M. Bellovin and Michael Merritt, “Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks”.
RFC 2463 Internet Control Message Protocol Dec. 1998.
RFC2246 The TLS Protocol, Jan. 1999.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2408 (“RFC 2408”) , 1998.
High Bandwidth Digital Content Protection System Feb. 17, 2000.
High Bandwidth Digital Content Protection System Revision 1.0 Erratum Mar. 1, 2001.
Digital Transmission Content Protection Specification vol. 1 Hitachi Ltd. Revision 1.0 Apr. 12, 1999.
Digital Transmission Content Protection Specification vol. 1 (Informational Version) Hitachi Ltd. Revision 1.2A Feb. 25, 2002.
Declaration of William Rosenblatt, Microsoft Exhibit 1009, 2018.
Supplemental Declaration of William Rosenblatt, Microsoft Exhibit 1015, 2018.
Petition for Inter Parties Review of U.S. Pat. No. 8,543,819, 2018.
Patent Owner’s Preliminary Response, 2018.
Petitioners’ Reply to Patent Owner’s Preliminary Response, 2018.
Patent Owner’s Sur-Reply to Petitioners’ Reply, 2018.
Petition for Inter Parties Review of U.S. Pat. No. 9,436,809, 2018.
Markman Order Filed Jul. 11, 2017.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2407 (“RFC 2407”), Nov. 1998.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2409 (“RFC 2409”), Nov. 1998.

* cited by examiner

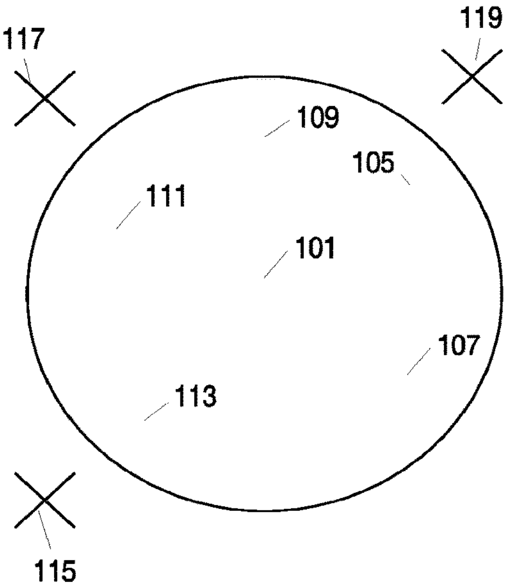

117
119
109
105
111
101
107
113
115

FIG. 1

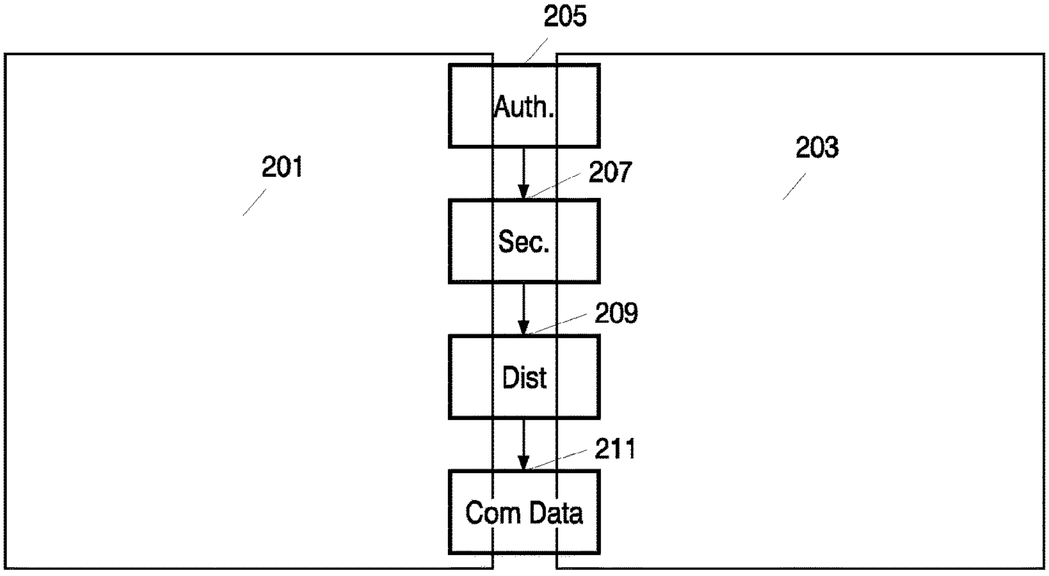

205
Auth.
201
207
203
Sec.
209
Dist
211
Com Data

FIG. 2

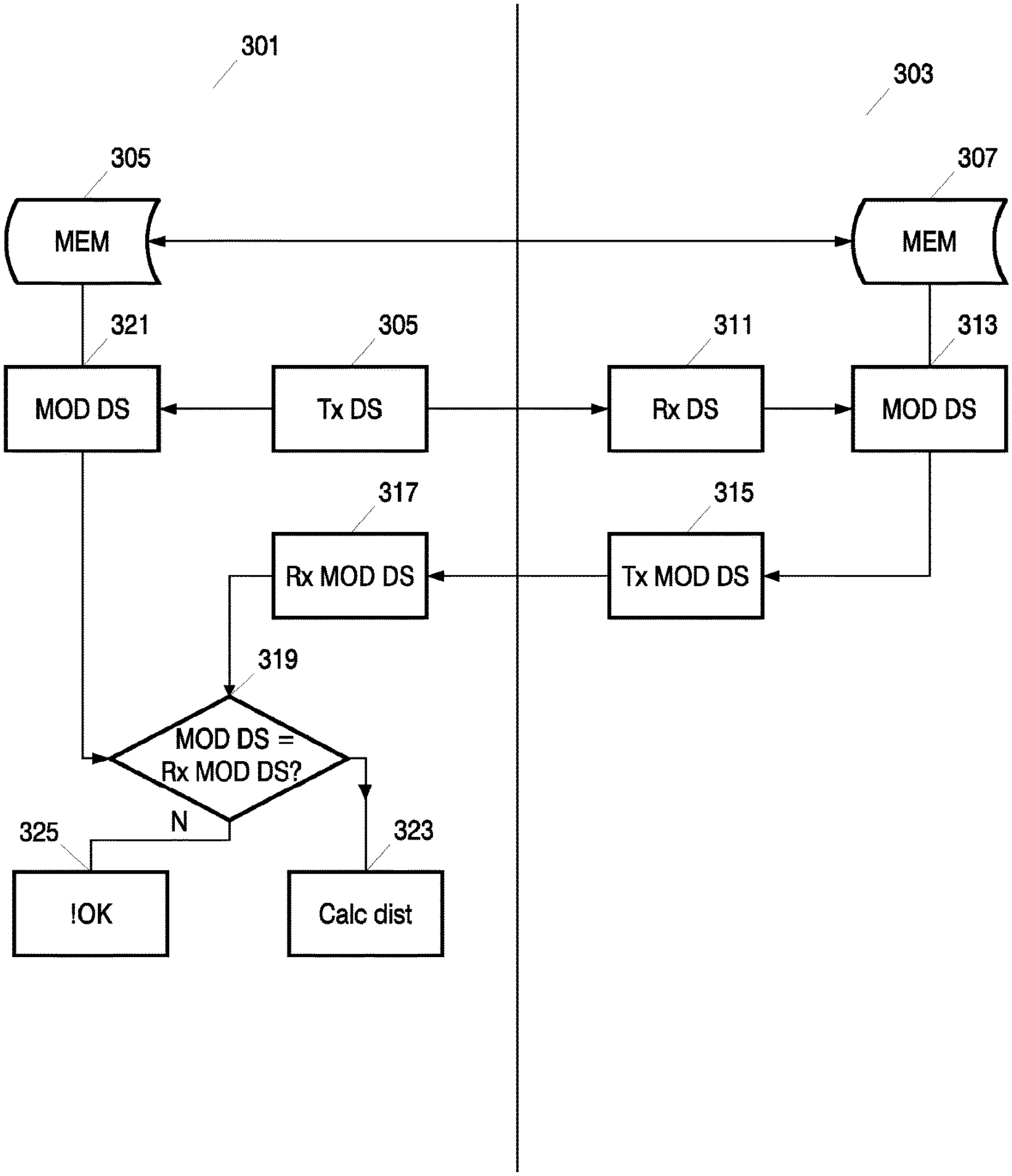

301
303
305
MEM
307
MEM
321
MOD DS
305
Tx DS
311
Rx DS
313
MOD DS
317
Rx MOD DS
315
Tx MOD DS
319
MOD DS =
Rx MOD DS?
N
325
!OK
323
Calc dist

FIG. 3

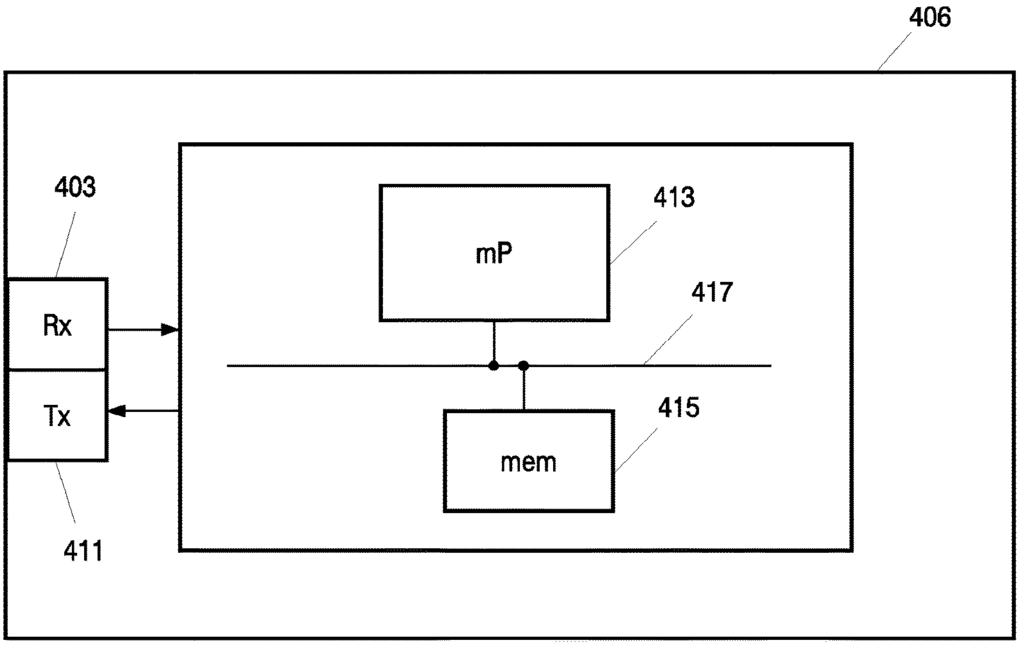


FIG. 4

SECURE AUTHENTICATED DISTANCE MEASUREMENT

This application is a continuation of the patent applications entitled "Secure Authenticated Distance Measurement", filed on Aug. 5, 2016 and afforded Ser. No. 15/229,207 which is a continuation of the application filed Nov. 11, 2014 and afforded Ser. No. 14/538,493 which claims priority pursuant to 35 USC 120, priority to and the benefit of the earlier filing date of, that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), which claimed priority to and the benefit of the earlier filing date, as a National Stage Filing of that international patent application filed on Jun. 27, 2003 and afforded serial number PCT/IB2003/02932 (WO2004014037), which claimed priority to and the benefit of the earlier filing date of that patent application filed on Jul. 26, 2002 and afforded serial number EP 02078076.3, the contents of all of which are incorporated by reference, herein.

This application is further related to that patent application entitled "Secure authenticated Distance Measurement", filed on Jul. 24, 2009 and afforded Ser. No. 12/508,917 (now U.S. Pat. No. 8,543,819), issued Sep. 24, 2013), which claimed priority to and the benefit of the earlier filing date of that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), the contents of which are incorporated by reference herein.

The invention relates to a method for a first communication device to performing authenticated distance measurement between a first communication device and a second communication device. The invention also relates to a method of determining whether data stored on a first communication device is to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

Digital media have become popular carriers for various types of data information. Computer software and audio information, for instance, are widely available on optical compact disks (CDs) and recently also DVD has gained in distribution share. The CD and the DVD utilize a common standard for the digital recording of data, software, images, and audio. Additional media, such as recordable discs, solid-state memory, and the like, are making considerable gains in the software and data distribution market.

The substantially superior quality of the digital format as compared to the analog format renders the former substantially more prone to unauthorized copying and pirating, further a digital format is both easier and faster to copy. Copying of a digital data stream, whether compressed, uncompressed, encrypted or non-encrypted, typically does not lead to any appreciable loss of quality in the data. Digital copying thus is essentially unlimited in terms of multi-generation copying. Analog data with its signal to noise ratio loss with every sequential copy, on the other hand, is naturally limited in terms of multi-generation and mass copying.

The advent of the recent popularity in the digital format has also brought about a slew of copy protection and DRM systems and methods. These systems and methods use technologies such as encryption, watermarking and right descriptions (e.g. rules for accessing and copying data).

One way of protecting content in the form of digital data is to ensure that content will only be transferred between devices if

- the receiving device has been authenticated as being a compliant device, and
- the user of the content has the right to transfer (move, copy) that content to another device.

If transfer of content is allowed, this will typically be performed in an encrypted way to make sure that the content cannot be captured illegally in a useful format.

Technology to perform device authentication and encrypted content transfer is available and is called a secure authenticated channel (SAC). Although it might be allowed to make copies of content over a SAC, the content industry is very bullish on content distribution over the Internet. This results in disagreement of the content industry on transferring content over interfaces that match well with the Internet, e.g. Ethernet.

Further, it should be possible for a user visiting his neighbor to watch a movie, which he owns, on the neighbor's big television screen. Typically, the content owner will disallow this, but it might become acceptable if it can be proved that a license holder of that movie (or a device that the license holder owns) is near that television screen.

It is therefore of interest to be able to include an authenticated distance measurement when deciding whether content should be accessed or copied by other devices.

In the article by Stefan Brands and David Chaum, "Distance-Bounding protocols", Eurocrypt '93 (1993), Pages 344-359, integration of distance-bounding protocols with public-key identification schemes is described. Here distance measurement is described based on time measurement using challenge and response bits and with the use of a commitment protocol. This does not allow authenticated device compliancy testing and is not efficient when two devices must also authenticate each other.

It is an object of the invention to obtain a solution to the problem of performing a secure transfer of content within a limited distance.

This is obtained by a method for a first communication device to performing authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device.

Because the common secret is being used for performing the distance measurement, it can be ensured that when measuring the distance from the first communication device to the second communication device, it is the distance between the right devices that is being measured.

The method combines a distance measurement protocol with an authentication protocol. This enables authenticated device compliancy testing and is efficient, because a secure channel is anyhow needed to enable secure communication between devices and a device can first be tested on compliancy before a distance measurement is executed.

In a specific embodiment, the authenticated distance measurement is performed according to the following steps,

transmitting a first signal from the first communication device to the second communication device at a first time t**1**, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal to the first device,

receiving the second signal at a second time t**2**,

checking if the second signal has been modified according to the common secret,

determining the distance between the first and the second communication device according to a time difference between t**1** and t**2**.

When measuring a distance by measuring the time difference between transmitting and receiving a signal and using a secret, shared between the first and the second communication device, for determining whether the returned signal really originated from the second communication device, the distance is measured in a secure authenticated way ensuring that the distance will not be measured to a third communication device (not knowing the secret). Using the shared secret for modifying the signal is a simple way to perform a secure authenticated distance measurement.

In a specific embodiment the first signal is a spread spectrum signal. Thereby a high resolution is obtained and it is possible to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In another embodiment the step of checking if the second signal has been modified according to the common secret is performed by the steps of,

generating a third signal by modifying the first signal according to the common secret,

comparing the third signal with the received second signal.

This method is an easy and simple way of performing the check, but it requires that both the first communication device and the second communication device know how the first signal is being modified using the common secret.

In a specific embodiment the first signal and the common secret are bit words and the second signal comprises information being generated by performing an XOR between the bit words. Thereby, it is a very simple operation that has to be performed, resulting in demand for few resources by both the first and the second communication device when performing the operation.

In an embodiment the common secret has been shared before performing the distance measurement, the sharing being performed by the steps of,

performing an authentication check from the first communication device on the second communication device by checking whether the second communication device is compliant with a set of predefined compliance rules,

if the second communication device is compliant, sharing the common secret by transmitting the secret to the second communication device.

This is a secure way of performing the sharing of the secret, ensuring that only devices being compliant with compliance rules can receive the secret. Further, the shared secret can afterwards be used for generating a SAC channel between the two devices. The secret could be shared using e.g. key transport mechanisms as described in ISO 11770-3. Alternatively, a key agreement protocol could be used, which e.g. is also described in ISO 11770-3.

In another embodiment the authentication check further comprises checking if the identification of the second device is compliant with an expected identification. Thereby, it is ensured that the second device really is the device that it should be. The identity could be obtained by checking a certificate stored in the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between the first and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. By using the authenticated distance measurement in connection with sharing data between devices, unauthorized distribution of content can be reduced.

In a specific embodiment the data stored on the first device is sent to the second device if it is determined that the data stored on the first device are to be accessed by the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between a third communication device and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. In this embodiment, the distance is not measured between the first communication device, on which the data are stored, and the second communication device. Instead, the distance is measured between a third communication device and the second communication device, where the third communication device could be personal to the owner of the content.

The invention also relates to a communication device for performing authenticated distance measurement to a second communication device, where the communication device shares a common secret with the second communication device and where the communication device comprises means for measuring the distance to the second device using the common secret.

In an embodiment the device comprises:

means for transmitting a first signal to a second communication device at a first time t**1**, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal,

means for receiving the second signal at a second time t**2**,

means for checking if the second signal has been modified according to the common secret, and

means for determining the distance between the first and the second communication device according to a time difference between t**1** and t**2**.

The invention also relates to an apparatus for playing back multimedia content comprising a communication device according to the above.

In the following preferred embodiments of the invention will be described referring to the figures, wherein:

FIG. **1** illustrates authenticated distance measurement being used for content protection,

FIG. **2** is a flow diagram illustrating the method of performing authenticated distance measurement,

FIG. **3** illustrates in further detail the step of performing the authenticated distance measurement shown in FIG. **2**,

FIG. **4** illustrates a communication device for performing authenticated distance measurement.

FIG. **1** illustrates an embodiment where authenticated distance measurement is being used for content protection. In the center of the circle **101** a computer **103** is placed. The computer comprises content, such as multimedia content being video or audio, stored on e.g. a hard disk, DVD or a CD. The owner of the computer owns the content and therefore the computer is authorized to access and present the multimedia content for the user. When the user wants to make a legal copy of the content to another device via e.g. a SAC, the distance between the other device and the computer **103** is measured and only devices within a predefined distance illustrated by the devices **105**, **107**, **109**, **111**, **113** inside the circle **101** are allowed to receive the content. Whereas the devices **115**, **117**, **119** having a distance to the computer **101** being larger than the predefined distance are not allowed to receive the content.

In the example a device is a computer, but it could e.g. also be a DVD drive, a CD drive or a Video, as long as the device comprises a communication device for performing the distance measurement.

In a specific example the distance might not have to be measured between the computer, on which the data are stored, and the other device, it could also be a third device e.g. a device being personal to the owner of the content which is within the predefined distance.

In FIG. **2** a flow diagram illustrates the general idea of performing authenticated distance measurement between two devices, **201** and **203** each comprising communication devices for performing the authenticated distance measurement. In the example the first device **201** comprises content which the second device **203** has requested. The authenticated distance measurement then is as follows. In step **205** the first device **201** authenticates the second device **203**; this could comprise the steps of checking whether the second device **203** is a compliant device and might also comprise the step of checking whether the second device **203** really is the device identified to the first device **201**. Then in step **207**, the first device **201** exchanges a secret with the second device **203**, which e.g. could be performed by transmitting a random generated bit word to second device **203**. The secret should be shared securely, e.g. according to some key management protocol as described in e.g. ISO 11770.

Then in step **209**, a signal for distance measurement is transmitted to the second device **203**; the second device modifies the received signal according to the secret and retransmits the modified signal back to the first device. The first device **201** measures the round trip time between the signal leaving and the signal returning and checks if the returned signal was modified according to the exchanged secret. The modification of the returned signal according to some secret will most likely be dependent on the transmission system and the signal used for distance measurement, i.e. it will be specific for each communication system (such as 1394, Ethernet, Bluetooth, IEEE 802.11, etc.).

The signal used for the distance measurement may be a normal data bit signal, but also special signals other than for data communication may be used. In an embodiment spread spectrum signals are used to be able to get high resolution and to be able to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In a specific example a direct sequence spread spectrum signal is used for distance measurement; this signal could be modified by XORing the chips (e.g. spreading code consisting of 127 chips) of the direct sequence code by the bits of the secret (e.g. secret consists also of 127 bits). Also, other mathematical operations as XOR could be used.

The authentication **205** and exchange of secret **207** could be performed using the protocols described in some known ISO standards ISO 9798 and ISO 11770. For example the first device **201** could authenticate the second device **203** according to the following communication scenario:

First device–>Second device: R_B||Text **1**
where R_B is a random number
Second device–>First device: CertA||TokenAB
Where CertA is a certificate of A
TokenAB=R_A||R_B||B||Text**3**||sS_A(R_A||R_B||B||Text**2**)
R_A is a random number
Indentifier B is an option
sS_A is a signature set by A using private key S_A

If TokenAB is replaced with the token as specified in ISO 11770-3 we at the same time can do secret key exchange. We can use this by substituting Text**2** by:

Text**2**:=eP_B(A||K||||Text**2**)||Text**3**
Where ePB is encrypted with Public key B
A is identifier of A
K is a secret to be exchanged

In this case the second device **203** determines the key (i.e. has key control), this is also called a key transport protocol, but also a key agreement protocol could be used. This may be undesirable in which case it can be reversed, such that the first device determines the key. A secret key has now been exchanged according to step **207** in FIG. **2**. Again, the secret key could be exchanged by e.g. a key transport protocol or a key agreement protocol.

After the distance has been measured in a secure authenticated way as described above content, data can be sent between the first and the second device in step **211** in FIG. **2**.

FIG. **3** illustrates in further detail the step of performing the authenticated distance measurement. As described above the first device **301** and the second device **303** have exchanged a secret; the secret is stored in the memory **305** of the first device and the memory **307** of the second device. In order to perform the distance measurement, a signal is transmitted to the second device via a transmitter **309**. The second device receives the signal via a receiver **311** and **313** modifies the signal by using the locally stored secret. The signal is modified according to rules known by the first device **301** and transmitted back to the first device **301** via a transmitter **315**. The first device **301** receives the modified signal via a receiver **317** and in **319** the received modified signal is compared to a signal, which has been modified locally. The local modification is performed in **321** by using the signal transmitted to the second device in transmitter **309** and then modifying the signal using the locally stored secret similar to the modification rules used by the second device. If the received modified signal and the locally modified signal are identical, then the received signal is authenticated and can be used for determining the distance between the first and the second device. If the two signals are not identical, then the received signal cannot be authenticated and can therefore not be used for measuring the distance as illustrated by **325**. In **323** the distance is calculated between the first and the second device; this could e.g. be performed by measuring the time, when the signal is transmitted by the transmitter **309** from the first device to the second device and measuring when the receiver **317** receives the signal from the second device. The time difference between transmittal time and receive time can then be used for determining the physical distance between the first device and the second device.

In FIG. **4** a communication device for performing authenticated distance measurement is illustrated. The device **401** comprises a receiver **403** and a transmitter **411**. The device further comprises means for performing the steps described above, which could be by executing software using a microprocessor **413** connected to memory **415** via a communication bus **417**. The communication device could then be placed inside devices such as a DVD, a computer, a CD, a CD recorder, a television and other devices for accessing protected content.

What is claimed is:

1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:

receive a second device certificate from the second device prior to sending a first signal;

provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;

receive a second signal from the second device after providing the first signal; and

provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,

wherein the secret is known by the first device.

2. The first device of claim **1**, wherein the secret is securely provided to the second device by the first device.

3. The first device of claim **2**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal, wherein the modifying requires the secret; and

determining that the modified first signal is identical to the second signal.

4. The first device of claim **3** wherein the secret comprises a first random number.

5. The first device of claim **4** wherein the secret is encrypted with a public key.

6. The first device of claim **5** wherein the first signal comprises a second random number.

7. The first device of claim **2**, wherein the second signal comprises the first signal modified by the secret.

8. The first device of claim **2**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal, wherein the modifying requires the secret; and

determining that the modified second signal is identical to the first signal.

9. The first device of claim **1**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal, wherein the modifying requires the secret; and

determining that the modified first signal is identical to the second signal.

10. The first device of claim **1**, wherein the predetermined time is based on a communication system associated with the first device.

11. The first device of claim **1**, further comprising instructions arranged to provide the secret to the second device.

12. The first device of claim **1**, wherein the second signal comprises the first signal modified by the secret.

13. The first device of claim **1** wherein the secret comprises a random number.

14. The first device of claim **1** wherein the secret is encrypted with a public key.

15. The first device of claim **1** wherein the first signal comprises a random number.

16. The first device of claim **1**, wherein the second signal comprises an XOR operation of the first signal with the secret.

17. The first device of claim **1**, further comprising instructions arranged to receive the secret from the second device.

18. The first device of claim **1**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal, wherein the modifying requires the secret; and

determining that the modified second signal is identical to the first signal.

19. A method of controlling delivery of protected content from a first device to a second device, the first device comprising a processor circuit the processor circuit arranged to execute instructions implementing the method, the method comprising:

receiving a second device certificate from the second device prior to sending a first signal;

providing the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;

receiving a second signal from the second device after providing the first signal;

sending the protected content from the first device to the second device when the second signal is derived from the secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,

wherein the secret is known by the first device.

20. The method of claim **19**, wherein the secret is securely provided to the second device by the first device.

21. The method of claim **20**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal according to the secret; and

determining that the modified first signal is identical to the second signal.

22. The method of claim **21**, wherein the secret comprises a first random number.

23. The method of claim **22**, wherein the secret is encrypted with a public key.

24. The method of claim **23**, wherein the first signal comprises a second random number.

25. The method of claim **20**, wherein the second signal comprises the first signal modified by the secret.

26. The method of claim **20**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal according to the secret; and

determining that the modified second signal is identical to the first signal.

27. The method of claim **19**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal according to the secret; and

determining that the modified first signal is identical to the second signal.

28. The method of claim **19**, wherein the predetermined time is based on a communication system associated with the first device.

29. The method of claim **19**, further comprising providing the secret to the second device.

30. The method of claim **19**, wherein the second signal comprises the first signal modified by the secret.

31. The method of claim **19**, wherein the secret comprises a random number.

32. The method of claim **19**, wherein the secret is encrypted with a public key.

33. The method of claim **19**, wherein the first signal comprises a random number.

34. The method of claim **19**, wherein the second signal comprises an XOR operation of the first signal with the secret.

35. The method of claim **19**, further comprising instructions arranged to receive the secret from the second device.

36. The method of claim **19**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal according to the secret; and

determining that the modified second signal is identical to the first signal.

* * * * *

EXHIBIT C

US010298564B2

(12) **United States Patent**
Kamperman

(10) **Patent No.: US 10,298,564 B2**
(45) **Date of Patent: *May 21, 2019**

(54) **SECURE AUTHENTICATED DISTANCE MEASUREMENT**

(71) Applicant: **KONINKLIJKE PHILIPS N.V.**, Eindhoven (NL)

(72) Inventor: **Franciscus L. A. J. Kamperman**, Geldrop (NL)

(73) Assignee: **KONINKLIJKE PHILIPS N.V.**, Eindhoven (NL)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **16/117,019**

(22) Filed: **Aug. 30, 2018**

(65) **Prior Publication Data**

US 2019/0014106 A1 Jan. 10, 2019

Related U.S. Application Data

(63) Continuation of application No. 15/352,646, filed on Nov. 16, 2016, now Pat. No. 10,091,186, which is a (Continued)

(30) **Foreign Application Priority Data**

Jul. 26, 2002 (EP) 02078076

(51) **Int. Cl.**
H04L 29/06 (2006.01)
H04L 9/14 (2006.01)
(Continued)

(52) **U.S. Cl.**
CPC ***H04L 63/0823*** (2013.01); ***G06F 21/10*** (2013.01); ***H04L 9/14*** (2013.01);
(Continued)

(58) **Field of Classification Search**
CPC H04L 63/0823; H04L 9/14; H04L 63/107; H04L 63/062; H04L 43/16;
(Continued)

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,438,824	A	3/1984	Mueller-Scholoer
4,688,036	A	8/1987	Hirano et al.

(Continued)

FOREIGN PATENT DOCUMENTS

EP	1100035	A1	5/2001
JP	H04306760	A	10/1992

(Continued)

OTHER PUBLICATIONS

Ikeno et al "Modern Cryptography Theory" Japan, Institute of Electronics, Information and Communication Engineers, Nov. 15, 1997, p. 175-177.

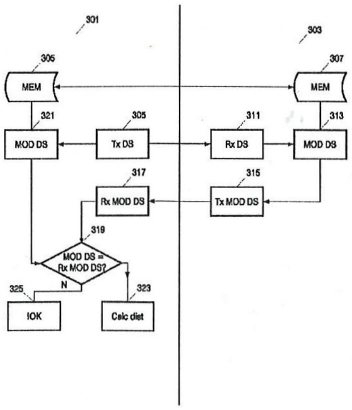
(Continued)

Primary Examiner — Darren B Schwartz

(57) **ABSTRACT**

The invention relates to a method for a first communication device to perform authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device. The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

53 Claims, 3 Drawing Sheets



Related U.S. Application Data

continuation of application No. 15/229,207, filed on Aug. 5, 2016, now Pat. No. 9,590,977, which is a continuation of application No. 14/538,493, filed on Nov. 11, 2014, now Pat. No. 9,436,809, which is a continuation of application No. 10/521,858, filed as application No. PCT/IB03/02932 on Jun. 27, 2003, now Pat. No. 8,886,939.

(51) **Int. Cl.**

H04L 12/26	(2006.01)
H04L 9/32	(2006.01)
G06F 21/10	(2013.01)
H04L 9/30	(2006.01)
H04W 24/00	(2009.01)
H04W 12/06	(2009.01)

(52) **U.S. Cl.**

CPC ***H04L 9/30*** (2013.01); ***H04L 9/3263*** (2013.01); ***H04L 43/0852*** (2013.01); ***H04L 43/16*** (2013.01); ***H04L 63/062*** (2013.01); ***H04L 63/107*** (2013.01); *G06F 2221/07* (2013.01); *G06F 2221/2111* (2013.01); *H04L 63/0428* (2013.01); *H04L 2463/101* (2013.01); *H04W 12/06* (2013.01); *H04W 24/00* (2013.01)

(58) **Field of Classification Search**

CPC H04L 43/0852; H04L 9/3263; H04L 9/30; H04L 63/0428; H04L 2463/101; G06F 21/10; G06F 2221/07; G06F 2221/2111; H04W 24/00; H04W 12/06

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,926,480	A	5/1990	Chaum	
5,126,746	A	6/1992	Gritton	
5,596,641	A	1/1997	Ohashi et al.	
5,602,917	A	2/1997	Mueller	
5,659,617	A	8/1997	Fischer	
5,723,911	A	3/1998	Glehr	
5,778,071	A	7/1998	Caputo et al.	
5,937,065	A	8/1999	Simon et al.	
5,949,877	A	9/1999	Traw et al.	
5,983,347	A	11/1999	Brinkmeyer et al.	
6,085,320	A	7/2000	Kaliski	
6,088,450	A	7/2000	Davis et al.	
6,151,676	A	11/2000	Cuccia et al.	
6,208,239	B1	3/2001	Muller et al.	
6,346,878	B1	2/2002	Pohlman et al.	
6,351,235	B1	2/2002	Stilp	
6,442,690	B1	8/2002	Howard, Jr.	
6,484,948	B1	11/2002	Sonoda	
6,493,825	B1	12/2002	Blumenau et al.	
6,526,598	B1	3/2003	Horn	
6,550,011	B1 *	4/2003	Sims, III	G06F 21/10 365/52
7,200,233	B1	4/2007	Keller et al.	
7,242,766	B1	7/2007	Lyle	
7,516,325	B2	4/2009	Willey	
7,787,865	B2	8/2010	Willey	
7,898,977	B2	3/2011	Roese	
8,068,610	B2	11/2011	Moroney	
8,107,627	B2	1/2012	Epstein	
8,352,582	B2	1/2013	Epstein	
8,997,243	B2	3/2015	Epstein	
2001/0008558	A1	7/2001	Hirafuji	
2001/0043702	A1	11/2001	Elteto et al.	
2001/0044786	A1	11/2001	Ishibashi	
2001/0050990	A1 *	12/2001	Sudia	G06Q 20/02 380/286
2002/0007452	A1 *	1/2002	Traw	G06F 21/10 713/152
2002/0026424	A1	2/2002	Akashi	
2002/0026576	A1	2/2002	Das-Purkayastha et al.	
2002/0035690	A1	3/2002	Nakano	
2002/0061748	A1	5/2002	Nakakita et al.	
2002/0078227	A1	6/2002	Kronenberg	
2002/0166047	A1	11/2002	Kawamoto	
2003/0021418	A1	1/2003	Arakawa et al.	
2003/0030542	A1	2/2003	Von Hoffmann	
2003/0051151	A1	3/2003	Asano	
2003/0065918	A1	4/2003	Willey	
2003/0070092	A1	4/2003	Hawkes et al.	
2003/0112978	A1	6/2003	Rodman et al.	
2003/0174838	A1 *	9/2003	Bremer	H04L 63/0428 380/270
2003/0184431	A1	10/2003	Lundkvist	
2003/0220765	A1	11/2003	Overy et al.	
2004/0015693	A1	1/2004	Kitazumi	
2004/0025018	A1 *	2/2004	Haas	H04L 45/26 713/168
2004/0080426	A1	4/2004	Fraenkel	
2005/0114647	A1	5/2005	Epstein	
2005/0265503	A1	12/2005	Rofheart et al.	
2006/0294362	A1	12/2006	Epstein	

FOREIGN PATENT DOCUMENTS

JP	H0619948	A	1/1994
JP	H08234658	A	9/1996
JP	9170364	A	6/1997
JP	H09170364	A	6/1997
JP	11101035	A	4/1999
JP	11208419	A	8/1999
JP	2000357156	A	12/2000
JP	2001249899	A	9/2001
JP	2001257672	A	9/2001
JP	2002124960		4/2002
JP	2002189966	A	7/2002
WO	9739553	A1	10/1997
WO	9949378		9/1999
WO	0152234	A1	7/2001
WO	0193434	A1	12/2001
WO	0233887	A2	4/2002
WO	0235036	A1	5/2002
WO	02054353	A1	7/2002

OTHER PUBLICATIONS

Modern Cryptography Theory (1986) Chapter 9, ISBN: 4-88552-064-9 (Japanese).

Hayashi et al Encryption and Authentication Program Module , Technical Paper (Japanese) NTT R&D vol. 44, No. 10 Oct. 1, 1995.

Stefan Brands and Devid Chaum "Distance Bounding Protocols" Eurocrypt '93, (1993) p. 344-359.

Tim Kindber & Kan Zhang "Context Authention Using Constrained Channels" pp. 1-8 , Apr. 16, 2001.

Hitachi Ltd., 5C Digital Transmission Content Protection White Paper Rev. 1.0 Jul. 14, 1998, p. 1013.

Boyd et al "Protocols for Authention and Key Establishment" Spring-Verlag, Sep. 17, 2003, p. 116-120, 195, 305.

High Bandwidth Digital Content Protection System Feb. 17, 2000.

High Bandwidth Digital Content Protection System Revision 1.0 Erratum Mar. 1, 2001.

Digital Transmission Content Protection Specification vol. 1 Hitachi Ltd. Revision 1.0 Apr. 12, 1999.

Digital Transmission Content Protection Specification vol. 1 (Informational Version) Hitachi Ltd. Revision 1.2A Feb. 25, 2002.

SmartRight™ Certification for FCC Approval for Use with the Broadcast Flag, Mar. 1, 2004.

SmartRight™ Copy Protection for System for Digital Home Networks, Deployment Process, CPTWG, Nov. 28, 2001.

SmartRight™ Copy Protection System for Digital Home Networks, CPTWG, May 24, 2001.

SmartRight™ Digital Broadcast Content Protection, Presentation to FCC, Apr. 2, 2004 (cited in litigation).

(56) **References Cited**

OTHER PUBLICATIONS

SmartRight™ Technical White Paper, Version 1.7, Jan. 2003 ("White Paper") (cited in litigation).
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2408 ("RFC 2408")—cited in litigation, Nov. 1998.
International Standard ISO/IEC 11770-3 (1st ed.) ("ISO 11770-3") , Nov. 1, 1999.
Scott Crosby, et al., "A Cryptanalysis of the High-bandwidth Digital Content Protection System " Computer and Communications Security, (2001).
SmartRight™ Specifications Sep. 26, 2001.
SmartRight™ Copy Protection System for Digital Home Networks, CPTWG, Jul. 11, 2001.
Bruce Schneier, Applied Cryptography (2d ed. 1996) ("Schneier").
Steven M. Bellovin and Michael Merritt, "Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks", 2002.
RFC 2463 Internet Control Message Protocol Dec. 1998.
RFC2246 the TLS Protocol, Jan. 1999.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2408 ("RFC 2408") , Nov. 1998.
Declaration of William Rosenblatt, Microsoft Exhibit 1009 , Dec. 8, 2017.
Supplemental Declaration of William Rosenblatt, Microsoft Exhibit 1015 , Apr. 20, 2018.
Petition for Inter Parties Review of USP 8543819 , Dec. 8, 2017.
Patent Owner's Preliminary Response , Mar. 13, 2018.
Petitioners' Reply to Patent Owner's Preliminary Response , Apr. 20, 2018.
Patent Owner's Sur-Reply to Petitioners' Reply , May 4, 2018.
Petition for Inter Parties Review of USP 9436809 , Dec. 8, 2017.
Markman Order Filed Jul. 11, 2017.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2407 ("RFC 2407") , Nov. 1998.
Internet Security Association and Key Management Protocol (ISAKMP), Request for Comments 2409 ("RFC 2409") , Nov. 1998.

* cited by examiner

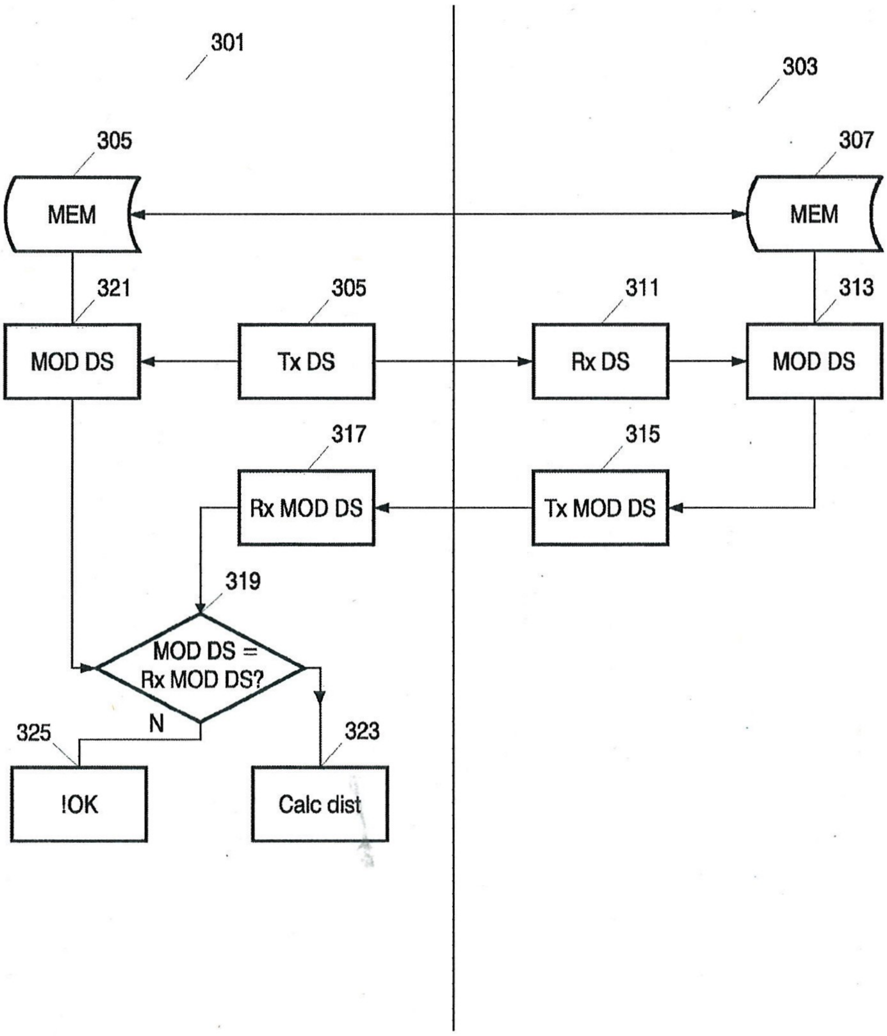

301
303
305
MEM
307
MEM
321
MOD DS
305
Tx DS
311
Rx DS
313
MOD DS
317
Rx MOD DS
315
Tx MOD DS
319
MOD DS = Rx MOD DS?
N
325
!OK
323
Calc dist

FIG. 3

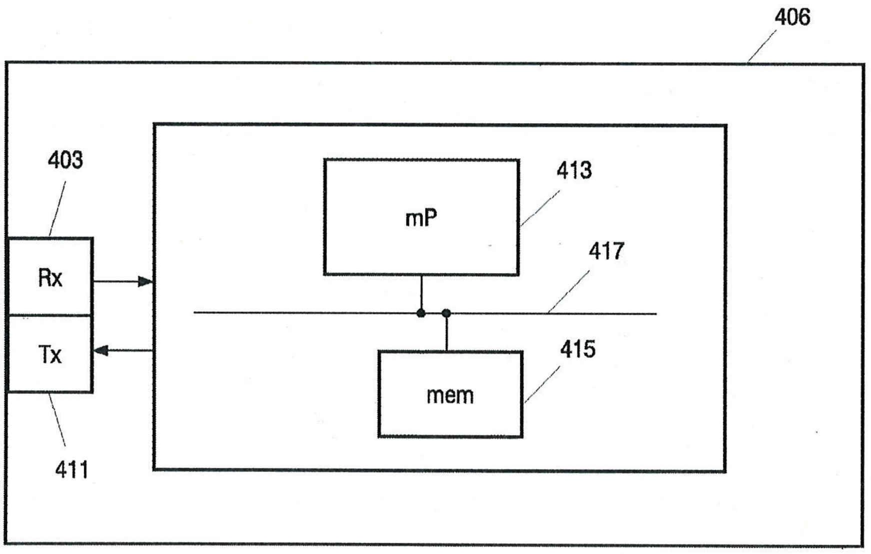


FIG. 4

SECURE AUTHENTICATED DISTANCE MEASUREMENT

This application is a continuation of the patent application entitled "Secure Authenticated Distance Measurement", filed on Nov. 16, 2016 and afforded Ser. No. 15/352,646 which is a continuation of the application filed Aug. 5, 2016 and afforded Ser. No. 15/229,207 which is a continuation of the application filed Nov. 11, 2014 and afforded Ser. No. 14/538,493 which claims priority pursuant to 35 USC 120, priority to and the benefit of the earlier filing date of, that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), which claimed priority to and the benefit of the earlier filing date, as a National Stage Filing of that international patent application filed on Jun. 27, 2003 and afforded serial number PCT/IB2003/02932 (WO2004014037), which claimed priority to and the benefit of the earlier filing date of that patent application filed on Jul. 26, 2002 and afforded serial number EP 02078076.3, the contents of all of which are incorporated by reference, herein.

This application is further related to that patent application entitled "Secure authenticated Distance Measurement", filed on Jul. 24, 2009 and afforded Ser. No. 12/508,917 (now U.S. Pat. No. 8,543,819), issued Sep. 24, 2013), which claimed priority to and the benefit of the earlier filing date of that patent application entitled "Secure Authenticated Distance Measurement", filed on Jan. 21, 2005 and afforded Ser. No. 10/521,858 (now U.S. Pat. No. 8,886,939), the contents of which are incorporated by reference herein.

The invention relates to a method for a first communication device to performing authenticated distance measurement between a first communication device and a second communication device. The invention also relates to a method of determining whether data stored on a first communication device is to be accessed by a second communication device. Moreover, the invention relates to a communication device for performing authenticated distance measurement to a second communication device. The invention also relates to an apparatus for playing back multimedia content comprising a communication device.

Digital media have become popular carriers for various types of data information. Computer software and audio information, for instance, are widely available on optical compact disks (CDs) and recently also DVD has gained in distribution share. The CD and the DVD utilize a common standard for the digital recording of data, software, images, and audio. Additional media, such as recordable discs, solid-state memory, and the like, are making considerable gains in the software and data distribution market.

The substantially superior quality of the digital format as compared to the analog format renders the former substantially more prone to unauthorized copying and pirating, further a digital format is both easier and faster to copy. Copying of a digital data stream, whether compressed, uncompressed, encrypted or non-encrypted, typically does not lead to any appreciable loss of quality in the data. Digital copying thus is essentially unlimited in terms of multi-generation copying. Analog data with its signal to noise ratio loss with every sequential copy, on the other hand, is naturally limited in terms of multi-generation and mass copying.

The advent of the recent popularity in the digital format has also brought about a slew of copy protection and DRM systems and methods. These systems and methods use technologies such as encryption, watermarking and right descriptions (e.g. rules for accessing and copying data).

One way of protecting content in the form of digital data is to ensure that content will only be transferred between devices if

- the receiving device has been authenticated as being a compliant device, and
- the user of the content has the right to transfer (move, copy) that content to another device.

If transfer of content is allowed, this will typically be performed in an encrypted way to make sure that the content cannot be captured illegally in a useful format.

Technology to perform device authentication and encrypted content transfer is available and is called a secure authenticated channel (SAC). Although it might be allowed to make copies of content over a SAC, the content industry is very bullish on content distribution over the Internet. This results in disagreement of the content industry on transferring content over interfaces that match well with the Internet, e.g. Ethernet.

Further, it should be possible for a user visiting his neighbor to watch a movie, which he owns, on the neighbor's big television screen. Typically, the content owner will disallow this, but it might become acceptable if it can be proved that a license holder of that movie (or a device that the license holder owns) is near that television screen.

It is therefore of interest to be able to include an authenticated distance measurement when deciding whether content should be accessed or copied by other devices.

In the article by Stefan Brands and David Chaum, "Distance-Bounding protocols", Eurocrypt '93 (1993), Pages 344-359, integration of distance-bounding protocols with public-key identification schemes is described. Here distance measurement is described based on time measurement using challenge and response bits and with the use of a commitment protocol. This does not allow authenticated device compliancy testing and is not efficient when two devices must also authenticate each other.

It is an object of the invention to obtain a solution to the problem of performing a secure transfer of content within a limited distance.

This is obtained by a method for a first communication device to performing authenticated distance measurement between the first communication device and a second communication device, wherein the first and the second communication device share a common secret and the common secret is used for performing the distance measurement between the first and the second communication device.

Because the common secret is being used for performing the distance measurement, it can be ensured that when measuring the distance from the first communication device to the second communication device, it is the distance between the right devices that is being measured.

The method combines a distance measurement protocol with an authentication protocol. This enables authenticated device compliancy testing and is efficient, because a secure channel is anyhow needed to enable secure communication between devices and a device can first be tested on compliancy before a distance measurement is executed.

In a specific embodiment, the authenticated distance measurement is performed according to the following steps,

- transmitting a first signal from the first communication device to the second communication device at a first time t1, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal to the first device,

receiving the second signal at a second time t2,

checking if the second signal has been modified according to the common secret,

determining the distance between the first and the second communication device according to a time difference between t1 and t2.

When measuring a distance by measuring the time difference between transmitting and receiving a signal and using a secret, shared between the first and the second communication device, for determining whether the returned signal really originated from the second communication device, the distance is measured in a secure authenticated way ensuring that the distance will not be measured to a third communication device (not knowing the secret). Using the shared secret for modifying the signal is a simple way to perform a secure authenticated distance measurement.

In a specific embodiment the first signal is a spread spectrum signal. Thereby a high resolution is obtained and it is possible to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In another embodiment the step of checking if the second signal has been modified according to the common secret is performed by the steps of,

generating a third signal by modifying the first signal according to the common secret,

comparing the third signal with the received second signal.

This method is an easy and simple way of performing the check, but it requires that both the first communication device and the second communication device know how the first signal is being modified using the common secret.

In a specific embodiment the first signal and the common secret are bit words and the second signal comprises information being generated by performing an XOR between the bit words. Thereby, it is a very simple operation that has to be performed, resulting in demand for few resources by both the first and the second communication device when performing the operation.

In an embodiment the common secret has been shared before performing the distance measurement, the sharing being performed by the steps of,

performing an authentication check from the first communication device on the second communication device by checking whether the second communication device is compliant with a set of predefined compliance rules,

if the second communication device is compliant, sharing the common secret by transmitting the secret to the second communication device.

This is a secure way of performing the sharing of the secret, ensuring that only devices being compliant with compliance rules can receive the secret. Further, the shared secret can afterwards be used for generating a SAC channel between the two devices. The secret could be shared using e.g. key transport mechanisms as described in ISO 11770-3. Alternatively, a key agreement protocol could be used, which e.g. is also described in ISO 11770-3.

In another embodiment the authentication check further comprises checking if the identification of the second device is compliant with an expected identification. Thereby, it is ensured that the second device really is the device that it should be. The identity could be obtained by checking a certificate stored in the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between the first and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. By using the authenticated distance measurement in connection with sharing data between devices, unauthorized distribution of content can be reduced.

In a specific embodiment the data stored on the first device is sent to the second device if it is determined that the data stored on the first device are to be accessed by the second device.

The invention also relates to a method of determining whether data stored on a first communication device are to be accessed by a second communication device, the method comprising the step of performing a distance measurement between a third communication device and the second communication device and checking whether the measured distance is within a predefined distance interval, wherein the distance measurement is an authenticated distance measurement according to the above. In this embodiment, the distance is not measured between the first communication device, on which the data are stored, and the second communication device. Instead, the distance is measured between a third communication device and the second communication device, where the third communication device could be personal to the owner of the content.

The invention also relates to a communication device for performing authenticated distance measurement to a second communication device, where the communication device shares a common secret with the second communication device and where the communication device comprises means for measuring the distance to the second device using the common secret.

In an embodiment the device comprises:

means for transmitting a first signal to a second communication device at a first time t1, the second communication device being adapted for receiving the first signal, generating a second signal by modifying the received first signal according to the common secret and transmitting the second signal,

means for receiving the second signal at a second time t2,

means for checking if the second signal has been modified according to the common secret, and

means for determining the distance between the first and the second communication device according to a time difference between t1 and t2.

The invention also relates to an apparatus for playing back multimedia content comprising a communication device according to the above.

In the following preferred embodiments of the invention will be described referring to the figures, wherein:

FIG. **1** illustrates authenticated distance measurement being used for content protection,

FIG. **2** is a flow diagram illustrating the method of performing authenticated distance measurement,

FIG. **3** illustrates in further detail the step of performing the authenticated distance measurement shown in FIG. **2**,

FIG. **4** illustrates a communication device for performing authenticated distance measurement.

FIG. **1** illustrates an embodiment where authenticated distance measurement is being used for content protection. In the center of the circle **101** a computer **103** is placed. The computer comprises content, such as multimedia content being video or audio, stored on e.g. a hard disk, DVD or a CD. The owner of the computer owns the content and therefore the computer is authorized to access and present the multimedia content for the user. When the user wants to make a legal copy of the content to another device via e.g. a SAC, the distance between the other device and the computer **103** is measured and only devices within a predefined distance illustrated by the devices **105**, **107**, **109**, **111**, **113** inside the circle **101** are allowed to receive the content. Whereas the devices **115**, **117**, **119** having a distance to the computer **101** being larger than the predefined distance are not allowed to receive the content.

In the example a device is a computer, but it could e.g. also be a DVD drive, a CD drive or a Video, as long as the device comprises a communication device for performing the distance measurement.

In a specific example the distance might not have to be measured between the computer, on which the data are stored, and the other device, it could also be a third device e.g. a device being personal to the owner of the content which is within the predefined distance.

In FIG. **2** a flow diagram illustrates the general idea of performing authenticated distance measurement between two devices, **201** and **203** each comprising communication devices for performing the authenticated distance measurement. In the example the first device **201** comprises content which the second device **203** has requested. The authenticated distance measurement then is as follows. In step **205** the first device **201** authenticates the second device **203**; this could comprise the steps of checking whether the second device **203** is a compliant device and might also comprise the step of checking whether the second device **203** really is the device identified to the first device **201**. Then in step **207**, the first device **201** exchanges a secret with the second device **203**, which e.g. could be performed by transmitting a random generated bit word to second device **203**. The secret should be shared securely, e.g. according to some key management protocol as described in e.g. ISO 11770.

Then in step **209**, a signal for distance measurement is transmitted to the second device **203**; the second device modifies the received signal according to the secret and retransmits the modified signal back to the first device. The first device **201** measures the round trip time between the signal leaving and the signal returning and checks if the returned signal was modified according to the exchanged secret. The modification of the returned signal according to some secret will most likely be dependent on the transmission system and the signal used for distance measurement, i.e. it will be specific for each communication system (such as 1394, Ethernet, Bluetooth, IEEE 802.11, etc.).

The signal used for the distance measurement may be a normal data bit signal, but also special signals other than for data communication may be used. In an embodiment spread spectrum signals are used to be able to get high resolution and to be able to cope with bad transmission conditions (e.g. wireless environments with a lot of reflections).

In a specific example a direct sequence spread spectrum signal is used for distance measurement; this signal could be modified by XORing the chips (e.g. spreading code consisting of 127 chips) of the direct sequence code by the bits of the secret (e.g. secret consists also of 127 bits). Also, other mathematical operations as XOR could be used.

The authentication **205** and exchange of secret **207** could be performed using the protocols described in some known ISO standards ISO 9798 and ISO 11770. For example the first device **201** could authenticate the second device **203** according to the following communication scenario:

First device→Second device: $R_B||\text{Text 1}$

where R_B is a random number

Second device→First device: CertA||TokenAB

Where CertA is a certificate of A

$\text{TokenAB}=R_A||R_B||B||\text{Text3}||sS_A(R_A||R_B||B||\text{Text2})$

R_A is a random number

Indentifier B is an option

sS_A is a signature set by A using private key S_A

If TokenAB is replaced with the token as specified in ISO 11770-3 we at the same time can do secret key exchange. We can use this by substituting Text2 by:

$\text{Text2}:=eP_B(A||K||\text{Text2})||\text{Text3}$

Where eP_B is encrypted with Public key B

A is identifier of A

K is a secret to be exchanged

In this case the second device **203** determines the key (i.e. has key control), this is also called a key transport protocol, but also a key agreement protocol could be used. This may be undesirable in which case it can be reversed, such that the first device determines the key. A secret key has now been exchanged according to step **207** in FIG. **2**. Again, the secret key could be exchanged by e.g. a key transport protocol or a key agreement protocol.

After the distance has been measured in a secure authenticated way as described above content, data can be sent between the first and the second device in step **211** in FIG. **2**.

FIG. **3** illustrates in further detail the step of performing the authenticated distance measurement. As described above the first device **301** and the second device **303** have exchanged a secret; the secret is stored in the memory **305** of the first device and the memory **307** of the second device. In order to perform the distance measurement, a signal is transmitted to the second device via a transmitter **309**. The second device receives the signal via a receiver **311** and **313** modifies the signal by using the locally stored secret. The signal is modified according to rules known by the first device **301** and transmitted back to the first device **301** via a transmitter **315**. The first device **301** receives the modified signal via a receiver **317** and in **319** the received modified signal is compared to a signal, which has been modified locally. The local modification is performed in **321** by using the signal transmitted to the second device in transmitter **309** and then modifying the signal using the locally stored secret similar to the modification rules used by the second device. If the received modified signal and the locally modified signal are identical, then the received signal is authenticated and can be used for determining the distance between the first and the second device. If the two signals are not identical, then the received signal cannot be authenticated and can therefore not be used for measuring the distance as illustrated by **325**. In **323** the distance is calculated between the first and the second device; this could e.g. be performed by measuring the time, when the signal is transmitted by the transmitter **309** from the first device to the second device and measuring when the receiver **317** receives the signal from the second device. The time difference between transmittal time and receive time can then be used for determining the physical distance between the first device and the second device.

In FIG. **4** a communication device for performing authenticated distance measurement is illustrated. The device **401** comprises a receiver **403** and a transmitter **411**. The device further comprises means for performing the steps described above, which could be by executing software using a microprocessor **413** connected to memory **415** via a communication bus **417**. The communication device could then be placed inside devices such as a DVD, a computer, a CD, a CD recorder, a television and other devices for accessing protected content.

The invention claimed is:

1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:
   - provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;
   - receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule;
   - create a second signal, wherein the second signal is derived from a secret known by the second device;
   - provide the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device; and
   - receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time.

2. The second device of claim 1, wherein the secret is securely provided to the second device by the first device.

3. The second device of claim 2, wherein determining that the second signal is derived from the secret comprises:
   - modifying the first signal, wherein the modifying requires the secret; and
   - determining that the modified first signal is identical to the second signal.

4. The second device of claim 2, wherein determining that the second signal is derived from the secret comprises:
   - modifying the first signal; and
   - determining that the modified first signal is identical to the second signal.

5. The second device of claim 2, wherein the predetermined time is based on a communication system associated with the first device.

6. The second device of claim 2, further comprising instructions arranged to receive the secret from the first device.

7. The second device of claim 2, wherein the second signal comprises the first signal modified by the secret.

8. The second device of claim 2, wherein the secret comprises a random number.

9. The second device of claim 2, wherein the secret is encrypted with a public key.

10. The second device of claim 2, wherein the first signal comprises a random number.

11. The second device of claim 2, wherein the second signal comprises an XOR operation of the first signal with the secret.

12. The second device of claim 2, wherein determining that the second signal is derived from the secret comprises:
    - modifying the second signal, wherein the modifying requires the secret; and
    - determining that the modified second signal is identical to the first signal.

13. The second device of claim 2, wherein determining that the second signal is derived from the secret comprises:
    - modifying the second signal; and
    - determining that the modified second signal is identical to the first signal.

14. The second device of claim 2, wherein the secret is used for generating a secure channel between the first device and the second device.

15. The second device of claim 1, wherein determining that the second signal is derived from the secret comprises:
    - modifying the first signal, wherein the modifying requires the secret; and
    - determining that the modified first signal is identical to the second signal.

16. The second device of claim 1, wherein determining that the second signal is derived from the secret comprises:
    - modifying the first signal; and
    - determining that the modified first signal is identical to the second signal.

17. The second device of claim 1, wherein the predetermined time is based on a communication system associated with the first device.

18. The second device of claim 1, further comprising instructions arranged to receive the secret from the first device.

19. The second device of claim 1, wherein the second signal comprises the first signal modified by the secret.

20. The second device of claim 1, wherein the secret comprises a random number.

21. The second device of claim 1, wherein the secret is encrypted with a public key.

22. The second device of claim 1, wherein the first signal comprises a random number.

23. The second device of claim 1, wherein the second signal comprises an XOR operation of the first signal with the secret.

24. The second device of claim 1, further comprising instructions arranged to provide the secret to the first device.

25. The second device of claim 1, wherein the secret is used for generating a secure channel between the first device and the second device.

26. The second device of claim 1, wherein determining that the second signal is derived from the secret comprises:
    - modifying the second signal, wherein the modifying requires the secret; and
    - determining that the modified second signal is identical to the first signal.

27. The second device of claim 1, wherein determining that the second signal is derived from the secret comprises:
    - modifying the second signal; and
    - determining that the modified second signal is identical to the first signal.

28. The second device of claim 1, wherein the secret is known by the first device.

29. A method of receiving a protected content sent from a first device to a second device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions implementing the method, the method comprising:
    - providing a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;
    - receiving the first signal form the first device when the certificate indicates that the second device is compliant with at least one compliance rule;
    - creating a second signal, wherein the second signal is derived from a secret known by the second device;
    - providing the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device;

receiving the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time.

30. The method of claim **29**, wherein the secret is securely provided to the second device by the first device.

31. The method of claim **30**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal, wherein the modifying requires the secret; and

determining that the modified first signal is identical to the second signal.

32. The method of claim **31**, wherein the second signal comprises an XOR operation of the first signal with the secret.

33. The method of claim **31**, wherein the secret comprises a first random number.

34. The method of claim **33**, wherein the secret is used for generating a secure channel between the first device and the second device.

35. The method of claim **33**, wherein the secret is encrypted with a public key.

36. The method of claim **35**, wherein the first signal comprises a second random number.

37. The method of claim **30**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal; and

determining that the modified first signal is identical to the second signal.

38. The method of claim **30**, wherein the second signal comprises the first signal modified by the secret.

39. The method of claim **30**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal, wherein the modifying requires the secret; and

determining that the modified second signal is identical to the first signal.

40. The method of claim **30**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal, wherein the modifying requires the secret; and

determining that the modified second signal is identical to the first signal.

41. The method of claim **29**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal, wherein the modifying requires the secret; and

determining that the modified first signal is identical to the second signal.

42. The method of claim **29**, wherein determining that the second signal is derived from the secret comprises:

modifying the first signal; and

determining that the modified first signal is identical to the second signal.

43. The method of claim **29**, wherein the predetermined time is based on a communication system associated with the first device.

44. The method of claim **29**, further comprising receiving the secret from the first device.

45. The method of claim **29**, wherein the second signal comprises the first signal modified by the secret.

46. The method of claim **29**, wherein the secret comprises a random number.

47. The method of claim **29**, wherein the secret is encrypted with a public key.

48. The method of claim **29**, wherein the first signal comprises a random number.

49. The method of claim **29**, wherein the second signal comprises an XOR operation of the first signal with the secret.

50. The method of claim **29**, further comprising providing the secret to the first device.

51. The method of claim **29**, wherein the secret is used for generating a secure channel between the first device and the second device.

52. The method of claim **29**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal, wherein the modifying requires the secret; and

determining that the modified second signal is identical to the first signal.

53. The method of claim **29**, wherein determining that the second signal is derived from the secret comprises:

modifying the second signal; and

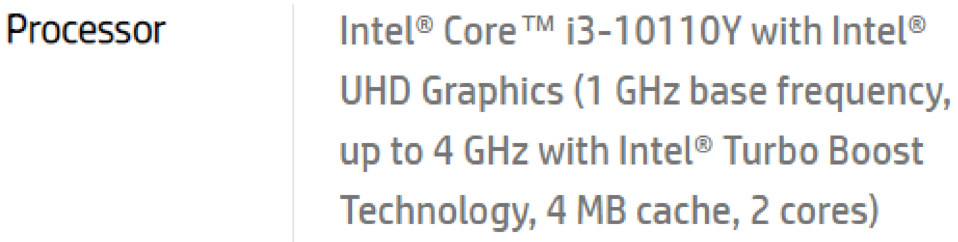
determining that the modified second signal is identical to the first signal.

* * * * *

EXHIBIT D

U.S. Patent No. 9,436,809

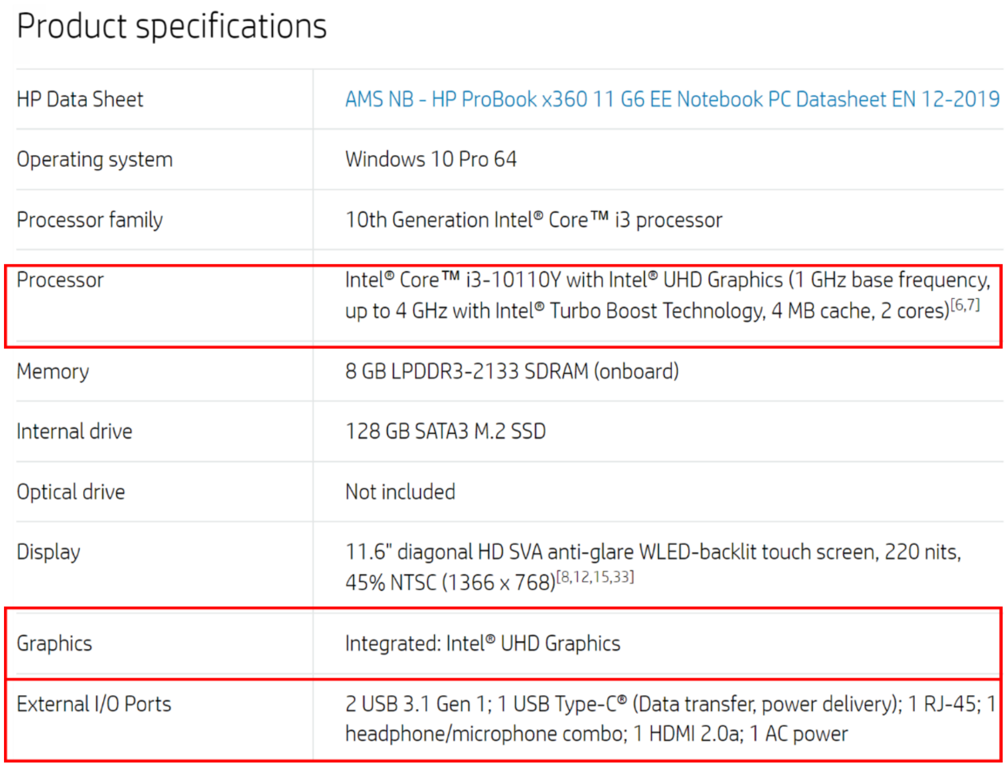
HP Product / Intel Product

	 **Processor** Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)
HP ProBook x360 11 G6 EE Notebook PC (Product # 3C534UT#ABA) (“HP Product” or “Accused Product”)	Intel video processing system and components thereof including 10th Generation Intel Core i3-10110Y Processor, main board hardware, integrated operating system, middleware, application program, video processing, and/or digital rights management (“DRM”) software that runs on the HP Product (“Intel Product” or “Accused Product”)

U.S. Patent No. 9,436,809: Claim 1

"1. A first device for controlling delivery of protected content to a second device, the first device comprising:"

1. A first device for controlling delivery of protected content to a second device, the first device comprising:	Each of the HP Product and the Intel Product is a first device for controlling delivery of protected content to a second device, and is referred to herein as an "Accused Product." For example, the HP Product is an HDMI transmitter with HDCP 2.2 for controlling delivery of protected content to another device, such as an HDMI receiver with HDCP 2.2.  HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. The HP Product includes an HDMI 2.0a port and a 10th Generation Intel® Core™ i3-10110Y Processor (the "Intel Processor") integrated with the Intel UHD Graphics 615 graphics processor (the "Intel GPU") that enable delivery of protected content to another device.

<table>
<tr>
<td></td>
<td>


Product specifications

<table>
<tr><td>HP Data Sheet</td><td>AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019</td></tr>
<tr><td>Operating system</td><td>Windows 10 Pro 64</td></tr>
<tr><td>Processor family</td><td>10th Generation Intel® Core™ i3 processor</td></tr>
<tr><td>Processor</td><td>Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]</td></tr>
<tr><td>Memory</td><td>8 GB LPDDR3-2133 SDRAM (onboard)</td></tr>
<tr><td>Internal drive</td><td>128 GB SATA3 M.2 SSD</td></tr>
<tr><td>Optical drive</td><td>Not included</td></tr>
<tr><td>Display</td><td>11.6" diagonal HD SVA anti-glare WLED-backlit touch screen, 220 nits, 45% NTSC (1366 x 768)[8,12,15,33]</td></tr>
<tr><td>Graphics</td><td>Integrated: Intel® UHD Graphics</td></tr>
<tr><td>External I/O Ports</td><td>2 USB 3.1 Gen 1; 1 USB Type-C® (Data transfer, power delivery); 1 RJ-45; 1 headphone/microphone combo; 1 HDMI 2.0a; 1 AC power</td></tr>
</table>

Id. *See also* NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.

The Intel Processor supports HDCP 2.2 via HDMI 2.0a.
</td>
</tr>
</table>

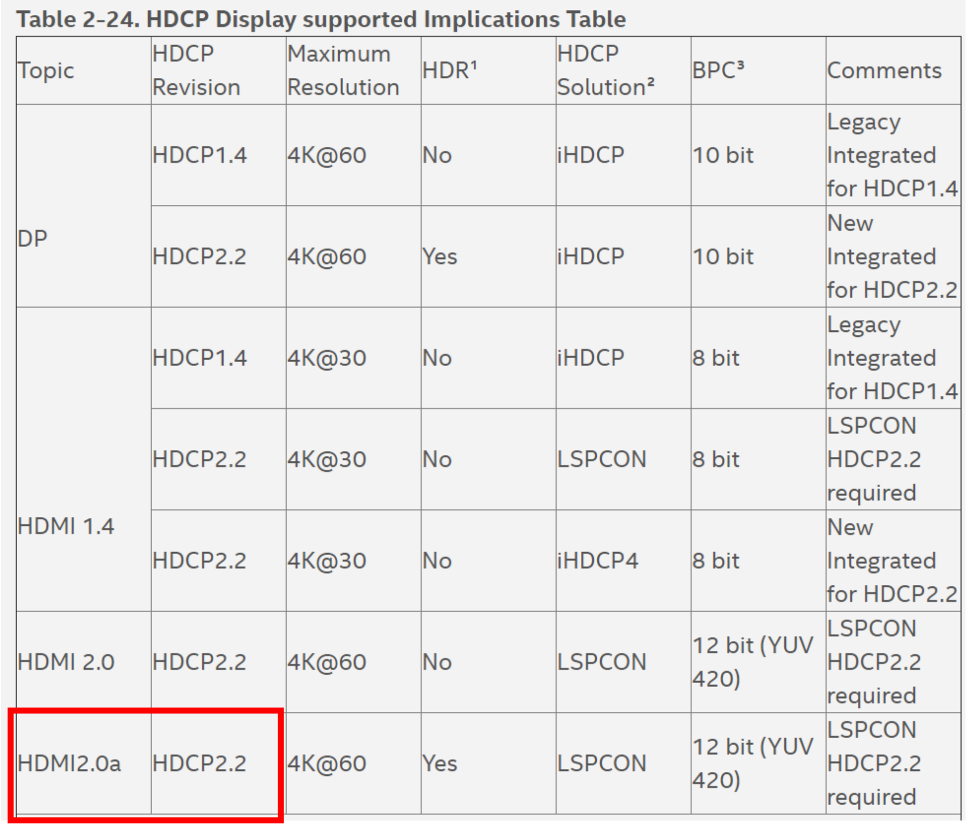
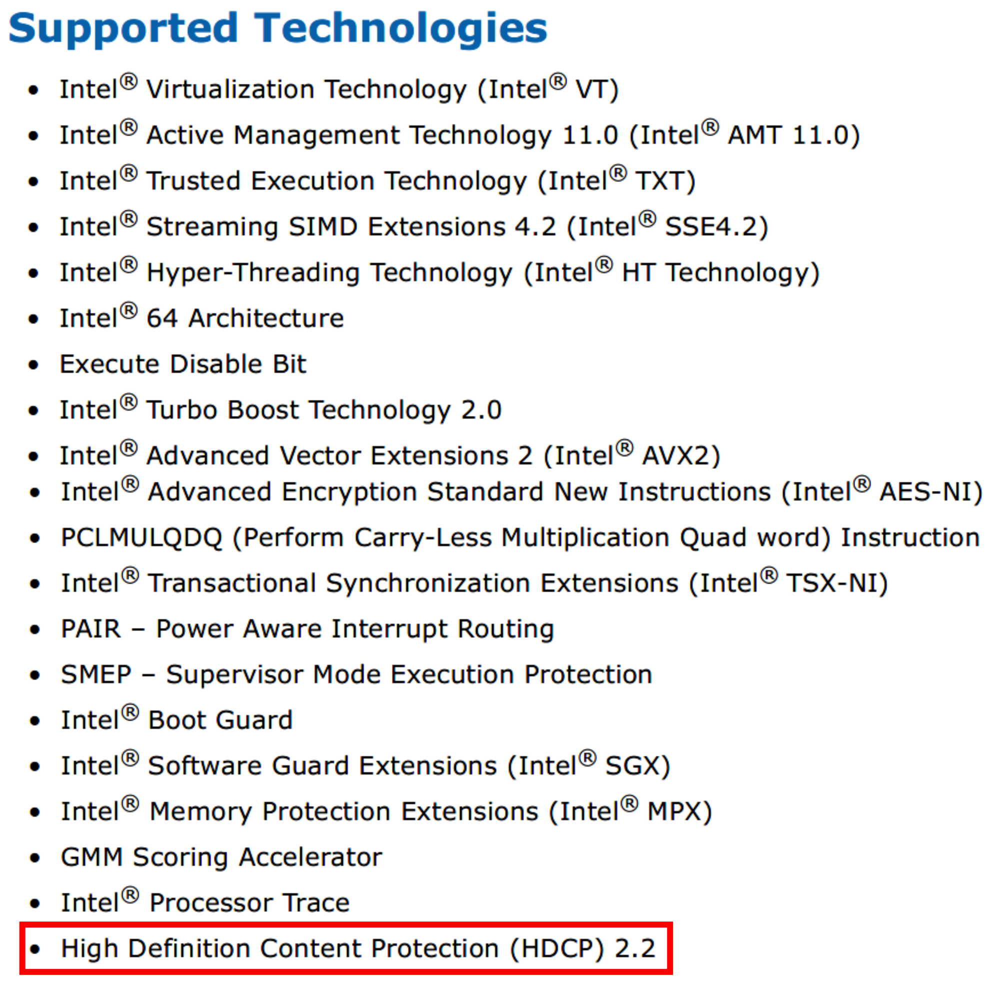


Table 2-24. HDCP Display supported Implications Table

Topic	HDCP Revision	Maximum Resolution	HDR¹	HDCP Solution²	BPC³	Comments
DP	HDCP1.4	4K@60	No	iHDCP	10 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@60	Yes	iHDCP	10 bit	New Integrated for HDCP2.2
HDMI 1.4	HDCP1.4	4K@30	No	iHDCP	8 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@30	No	LSPCON	8 bit	LSPCON HDCP2.2 required
	HDCP2.2	4K@30	No	iHDCP4	8 bit	New Integrated for HDCP2.2
HDMI 2.0	HDCP2.2	4K@60	No	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required
HDMI2.0a	HDCP2.2	4K@60	Yes	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required

Intel, How to enable High Dynamic Range?, https://www.intel.com/content/www/us/en/support/articles/000032112/graphics/graphics-for-7th-generation-intel-processors.html.

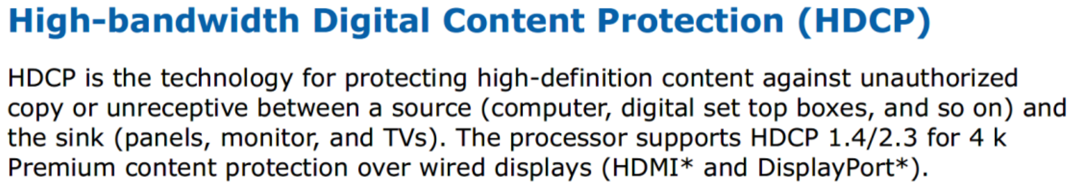


Supported Technologies

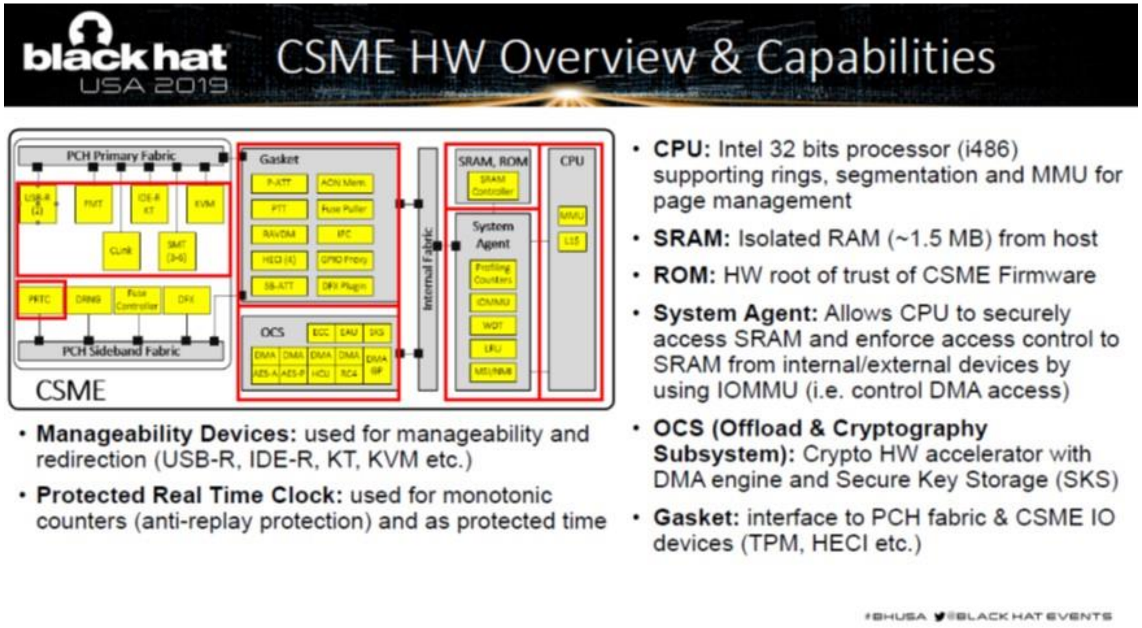
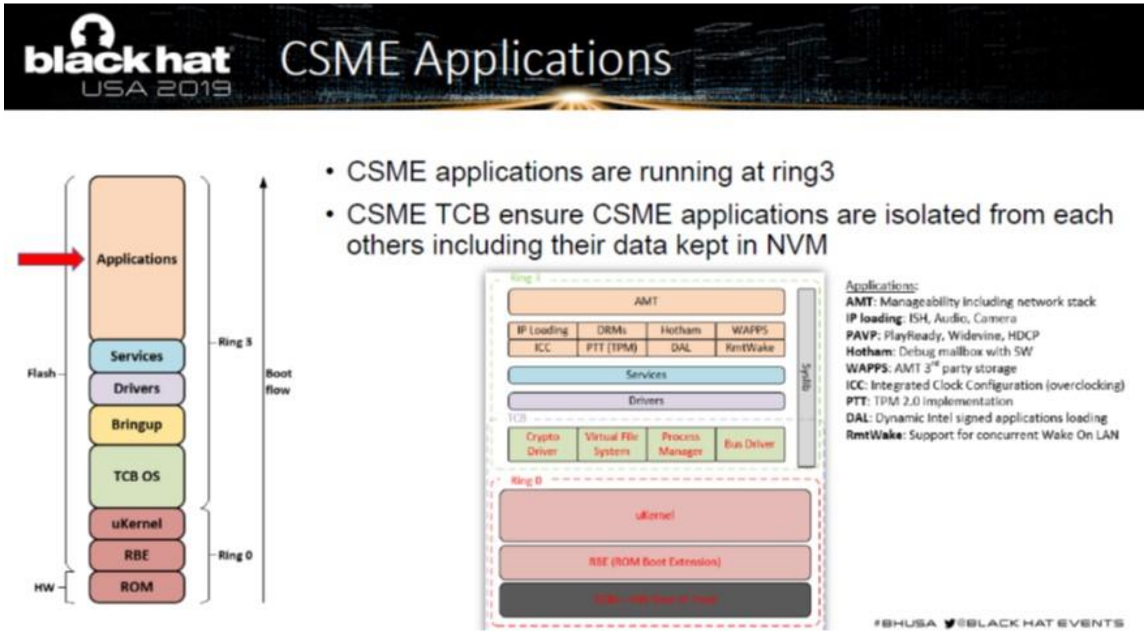
- Intel® Virtualization Technology (Intel® VT)
- Intel® Active Management Technology 11.0 (Intel® AMT 11.0)
- Intel® Trusted Execution Technology (Intel® TXT)
- Intel® Streaming SIMD Extensions 4.2 (Intel® SSE4.2)
- Intel® Hyper-Threading Technology (Intel® HT Technology)
- Intel® 64 Architecture
- Execute Disable Bit
- Intel® Turbo Boost Technology 2.0
- Intel® Advanced Vector Extensions 2 (Intel® AVX2)
- Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)
- PCLMULQDQ (Perform Carry-Less Multiplication Quad word) Instruction
- Intel® Transactional Synchronization Extensions (Intel® TSX-NI)
- PAIR – Power Aware Interrupt Routing
- SMEP – Supervisor Mode Execution Protection
- Intel® Boot Guard
- Intel® Software Guard Extensions (Intel® SGX)
- Intel® Memory Protection Extensions (Intel® MPX)
- GMM Scoring Accelerator
- Intel® Processor Trace
- High Definition Content Protection (HDCP) 2.2

Intel, 10th Generation Intel Core Processors, Datasheet, Volume 1 or 2 (Jul. 2020, rev. 5), *available at* https://cdrdv2.intel.com/v1/dl/getContent/615211, at 11-12.

“HDCP is the technology for protecting high-definition content against unauthorized copy … between a source … and the sink …. The [Intel] [P]rocessor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*).”

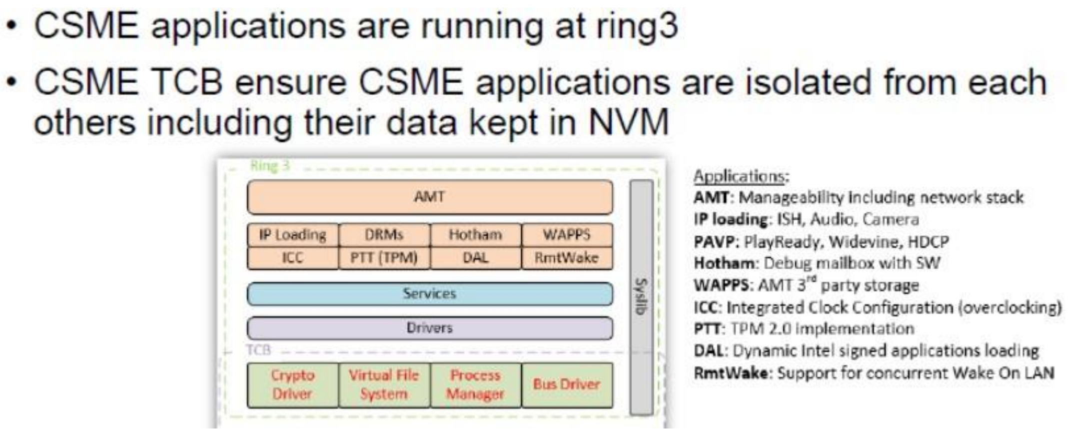
	 **High-bandwidth Digital Content Protection (HDCP)** HDCP is the technology for protecting high-definition content against unauthorized copy or unreceptive between a source (computer, digital set top boxes, and so on) and the sink (panels, monitor, and TVs). The processor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*). *Id.* at 44 Intel's "UHD" processor nomenclature also indicates support for HDCP 2.2: > Another change from 7 Gen to 8 Gen will be in the graphics. Intel is upgrading the nomenclature of the integrated graphics from HD 620 to UHD 620, indicating that the silicon is suited for 4K playback and processing. During our pre-briefing it was categorically stated several times that there was no change between the two, however we have since confirmed that the new chips will come with HDCP 2.2 support as standard for DP1.2a, removing the need for an external LSPCON for this feature. Other than this display controller change however, it appears that these new UHD iGPUs are architecturally the same as their HD predecessors. AnandTech, https://www.anandtech.com/show/11738/intel-launches-8th-generation-cpus-starting-with-kaby-lake-refresh-for-15w-mobile. HDCP 2.2 is implemented in Intel-based systems with Core-i series Processors within the Converged Security & Manageability Engine (CSME) also known as the Management Engine (ME). The CSME contains a processor (x86 core) which executes instructions including but not limited to the uKernel/OS, drivers, services, and applications for the CSME.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising:"

	 **CSME HW Overview & Capabilities** - **Manageability Devices:** used for manageability and redirection (USB-R, IDE-R, KT, KVM etc.) - **Protected Real Time Clock:** used for monotonic counters (anti-replay protection) and as protected time - **CPU:** Intel 32 bits processor (i486) supporting rings, segmentation and MMU for page management - **SRAM:** Isolated RAM (~1.5 MB) from host - **ROM:** HW root of trust of CSME Firmware - **System Agent:** Allows CPU to securely access SRAM and enforce access control to SRAM from internal/external devices by using IOMMU (i.e. control DMA access) - **OCS (Offload & Cryptography Subsystem):** Crypto HW accelerator with DMA engine and Secure Key Storage (SKS) - **Gasket:** interface to PCH fabric & CSME IO devices (TPM, HECI etc.) #BHUSA @BLACK HAT EVENTS Behind the Scenes of Intel Security and Manageability Engine, blackhat USA 2019 ("CSME") at 7.  **CSME Applications** - CSME applications are running at ring3 - CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM Applications: **AMT**: Manageability including network stack **IP loading**: ISH, Audio, Camera **PAVP**: PlayReady, Widevine, HDCP **Hotham**: Debug mailbox with SW **WAPPS**: AMT 3rd party storage **ICC**: Integrated Clock Configuration (overclocking) **PTT**: TPM 2.0 implementation **DAL**: Dynamic Intel signed applications loading **RmtWake**: Support for concurrent Wake On LAN #BHUSA @BLACK HAT EVENTS

Id. at 23.

One such application is “PAVP” which provides HDCP capabilities within the Intel processor.



- CSME applications are running at ring3
- CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM

Applications:
AMT: Manageability including network stack
IP loading: ISH, Audio, Camera
PAVP: PlayReady, Widevine, HDCP
Hotham: Debug mailbox with SW
WAPPS: AMT 3rd party storage
ICC: Integrated Clock Configuration (overclocking)
PTT: TPM 2.0 implementation
DAL: Dynamic Intel signed applications loading
RmtWake: Support for concurrent Wake On LAN

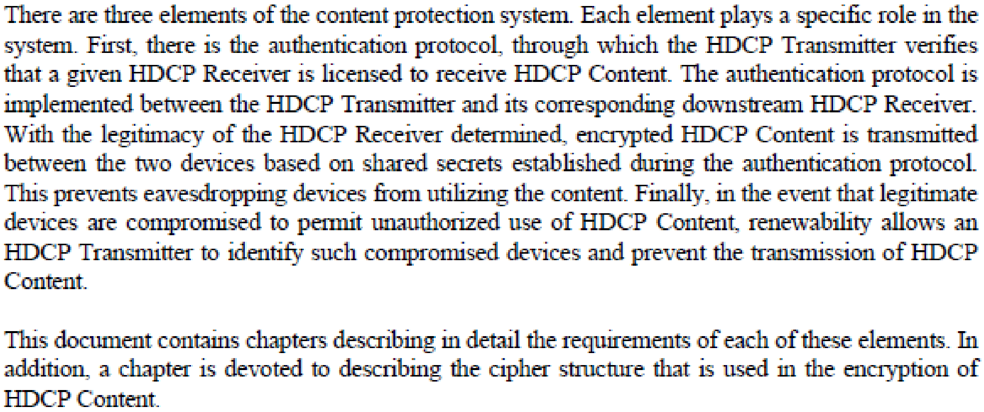
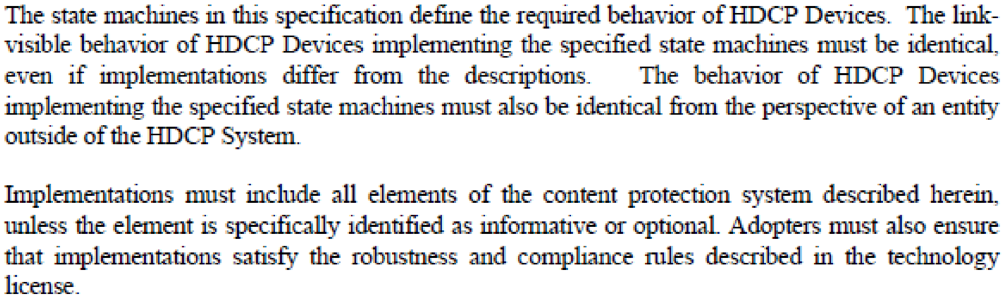
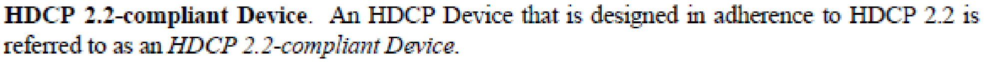
Id.

Upon information and belief, the Accused Product is compliant with the High-bandwidth Digital Content Protection System Revision 2.2 (“HDCP 2.2”) protocol. The Accused Product supports HDCP 2.2 for protecting content between devices.

> For the purpose of this specification, it is assumed that the Audiovisual content is transmitted over a HDMI based wired display link. In an HDCP System, two or more HDCP Devices are interconnected through an HDCP-protected Interface. The Audiovisual Content flows from the Upstream Content Control Function into the HDCP System at the most upstream HDCP Transmitter. From there the Audiovisual Content encrypted by the HDCP System, referred to as HDCP Content, flows through a tree-shaped topology of HDCP Receivers over HDCP-protected Interfaces. This specification describes a content protection mechanism for: (1) authentication of HDCP Receivers to their immediate upstream connection (i.e., an HDCP Transmitter), (2) revocation of HDCP Receivers that are determined by the Digital Content Protection, LLC, to be invalid, and (3) HDCP Encryption of Audiovisual Content over the HDCP-protected Interfaces between HDCP Transmitters and their downstream HDCP Receivers. HDCP Receivers may render the HDCP Content in audio and visual form for human consumption. HDCP Receivers may be HDCP Repeaters that serve as downstream HDCP Transmitters emitting the HDCP Content further downstream to one or more additional HDCP Receivers.

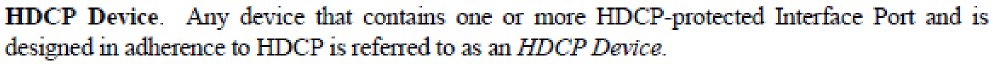
High-bandwidth Digital Content Protection System Mapping HDCP to HDMI Revision 2.2 13 February, 2013 (“HDMI HDCP 2.2”) at 5.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising:"

	 There are three elements of the content protection system. Each element plays a specific role in the system. First, there is the authentication protocol, through which the HDCP Transmitter verifies that a given HDCP Receiver is licensed to receive HDCP Content. The authentication protocol is implemented between the HDCP Transmitter and its corresponding downstream HDCP Receiver. With the legitimacy of the HDCP Receiver determined, encrypted HDCP Content is transmitted between the two devices based on shared secrets established during the authentication protocol. This prevents eavesdropping devices from utilizing the content. Finally, in the event that legitimate devices are compromised to permit unauthorized use of HDCP Content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of HDCP Content. This document contains chapters describing in detail the requirements of each of these elements. In addition, a chapter is devoted to describing the cipher structure that is used in the encryption of HDCP Content. *Id*. at 9. The Accused Product is an HDCP Device, and more specifically an HDCP 2.2-compliant Device, capable of functioning as an HDCP Transmitter and that implements required functionality of HDMI HDCP 2.2 including the functions required by the HDCP Transmitter State Diagram.  The state machines in this specification define the required behavior of HDCP Devices. The link-visible behavior of HDCP Devices implementing the specified state machines must be identical, even if implementations differ from the descriptions. The behavior of HDCP Devices implementing the specified state machines must also be identical from the perspective of an entity outside of the HDCP System. Implementations must include all elements of the content protection system described herein, unless the element is specifically identified as informative or optional. Adopters must also ensure that implementations satisfy the robustness and compliance rules described in the technology license. *Id*. at 5.  **HDCP 2.2-compliant Device**. An HDCP Device that is designed in adherence to HDCP 2.2 is referred to as an *HDCP 2.2-compliant Device*. *Id*. at 6.

U.S. Patent No. 9,436,809: Claim 1

"1. A first device for controlling delivery of protected content to a second device, the first device comprising:"



HDCP Device. Any device that contains one or more HDCP-protected Interface Port and is designed in adherence to HDCP is referred to as an *HDCP Device*.

Id. at 7.

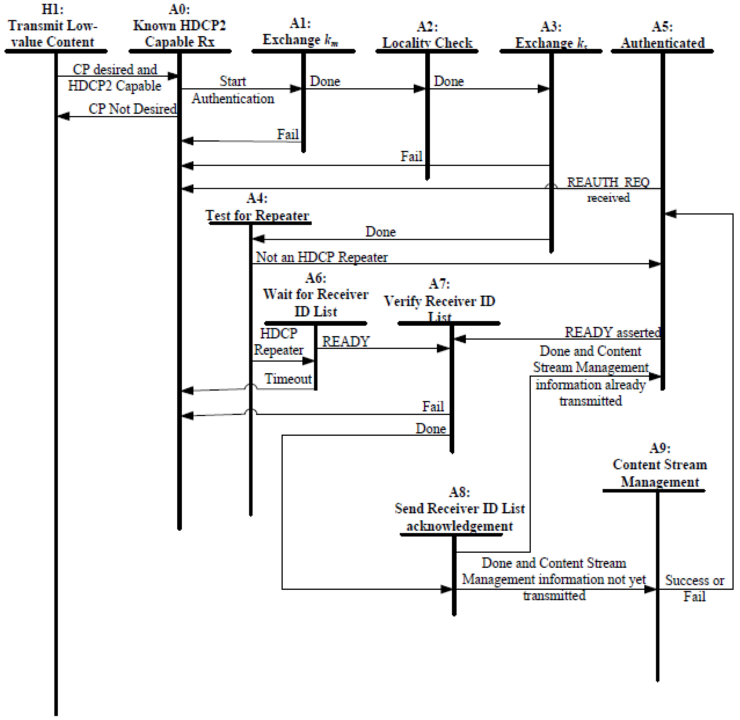
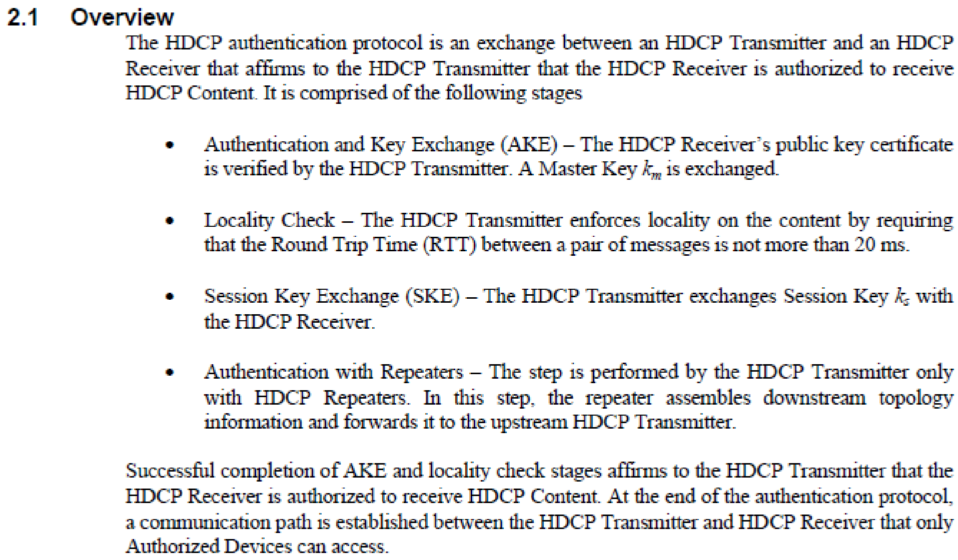


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27-30.

The Accused Product controls delivery of protected content to a second device.

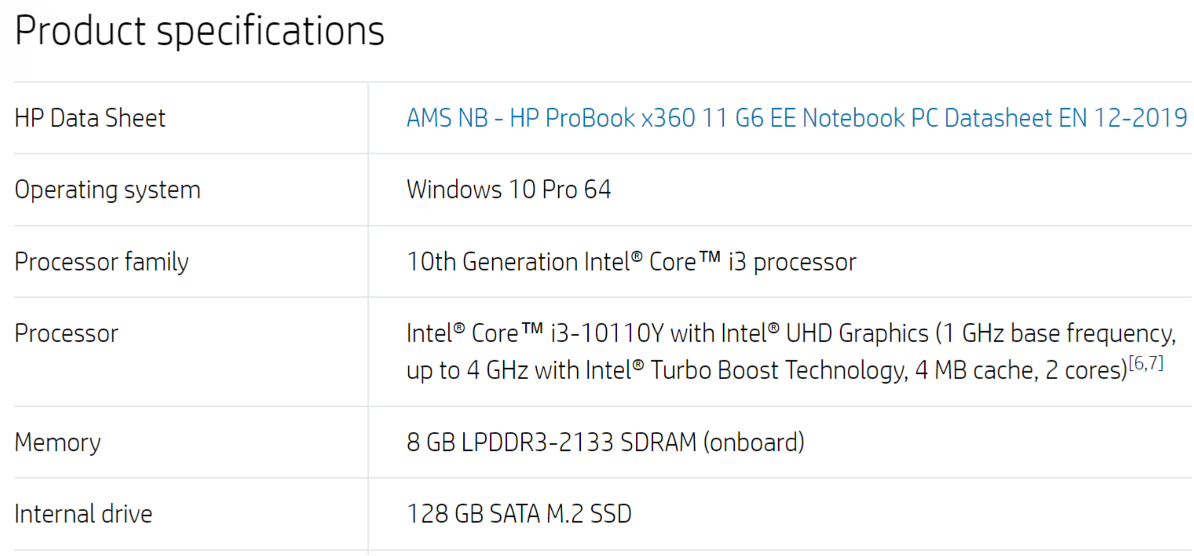
"1. A first device for controlling delivery of protected content to a second device, the first device comprising:"

	 **2.1 Overview** The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. *Id*. at 11.

"a memory;"

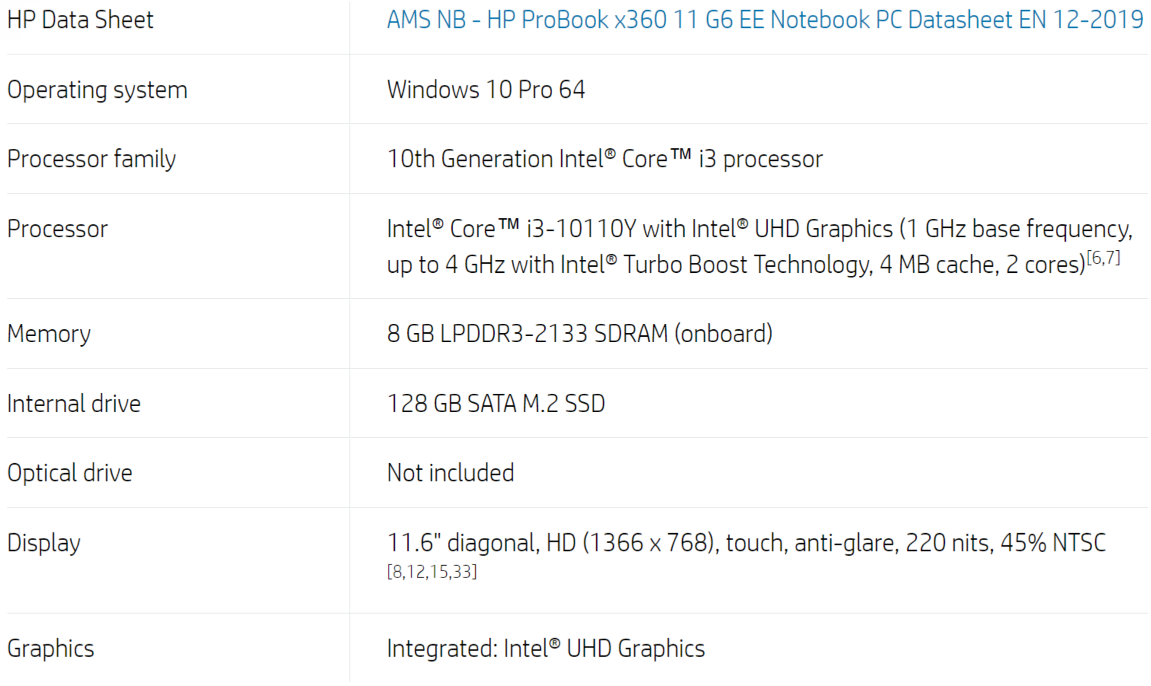
<table>
<tr>
<td>a memory;</td>
<td>
The Accused Product includes a memory.

For example, the Intel Processor includes a 4MB cache and the Accused Product also includes an 8GB onboard LPDDR3 memory in addition to a 128GB solid state hard drive.

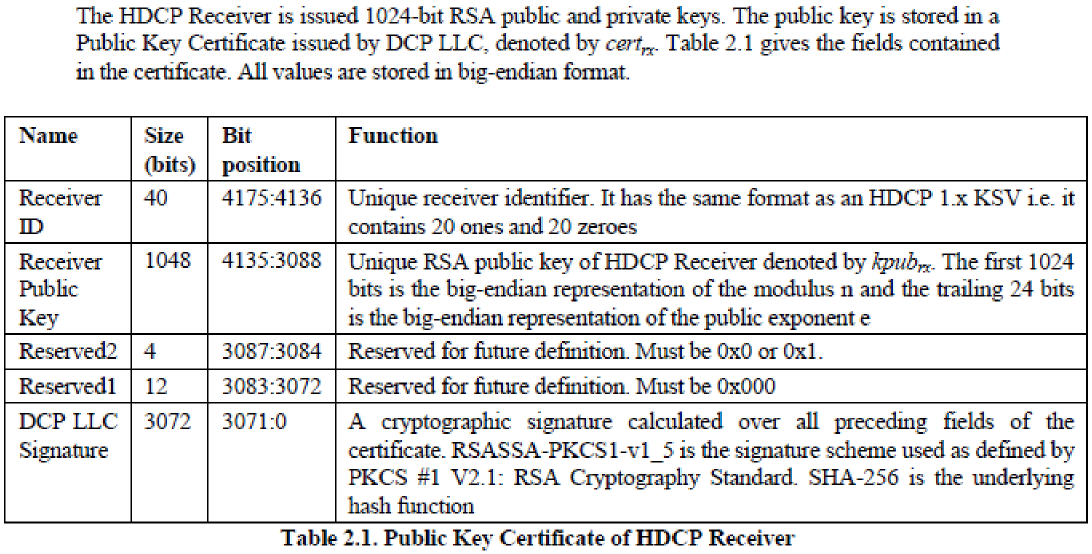
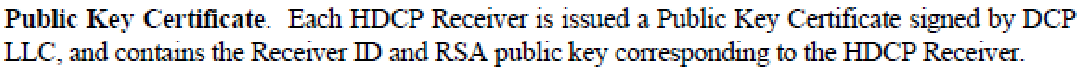

Product specifications
<table>
<tr><td>HP Data Sheet</td><td>AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019</td></tr>
<tr><td>Operating system</td><td>Windows 10 Pro 64</td></tr>
<tr><td>Processor family</td><td>10th Generation Intel® Core™ i3 processor</td></tr>
<tr><td>Processor</td><td>Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]</td></tr>
<tr><td>Memory</td><td>8 GB LPDDR3-2133 SDRAM (onboard)</td></tr>
<tr><td>Internal drive</td><td>128 GB SATA M.2 SSD</td></tr>
</table>

HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc.
</td>
</tr>
</table>

"a processor, said processor arranged to:"

<table>
<tr>
<td>a processor, said processor arranged to:</td>
<td>
<p>The Accused Product includes a processor.</p>
<p>For example, the Accused Product includes the Intel Processor integrated with the Intel GPU.</p>

<table>
<tr><td>HP Data Sheet</td><td>AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019</td></tr>
<tr><td>Operating system</td><td>Windows 10 Pro 64</td></tr>
<tr><td>Processor family</td><td>10th Generation Intel® Core™ i3 processor</td></tr>
<tr><td>Processor</td><td>Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]</td></tr>
<tr><td>Memory</td><td>8 GB LPDDR3-2133 SDRAM (onboard)</td></tr>
<tr><td>Internal drive</td><td>128 GB SATA M.2 SSD</td></tr>
<tr><td>Optical drive</td><td>Not included</td></tr>
<tr><td>Display</td><td>11.6" diagonal, HD (1366 x 768), touch, anti-glare, 220 nits, 45% NTSC [8,12,15,33]</td></tr>
<tr><td>Graphics</td><td>Integrated: Intel® UHD Graphics</td></tr>
</table>
<p>HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. <i>See also</i> NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.</p>
</td>
</tr>
</table>

"receive a certificate of the second device, the certificate providing information regarding the second device;"

<table>
<tr>
<td>receive a certificate of the second device, the certificate providing information regarding the second device;</td>
<td>
<p>The processor of the Accused Product is arranged to receive a certificate of the second device, e.g., $cert_{rx}$, as part of the Authentication and Key Exchange (AKE) stage of the HDCP 2.2 protocol, the certificate providing information regarding the second device.</p>
<p>The certificate, $cert_{rx}$, includes a Receiver ID for the second device, Receiver Public Key for the second device, and a cryptographic signature, amongst other information.</p>

<p>The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.</p>
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
<p>Table 2.1. Public Key Certificate of HDCP Receiver</p>
<p>HDMI HDCP 2.2 at 11.</p>

<p>Public Key Certificate. Each HDCP Receiver is issued a Public Key Certificate signed by DCP LLC, and contains the Receiver ID and RSA public key corresponding to the HDCP Receiver.</p>
<p>Id. at 8.</p>
<p>The Accused Product receives the certificate from the second device as part of the AKE stage, irrespective of whether the Accused Product has a Master Key k_m stored corresponding to the Receiver ID.</p>
</td>
</tr>
</table>

"receive a certificate of the second device, the certificate providing information regarding the second device;"

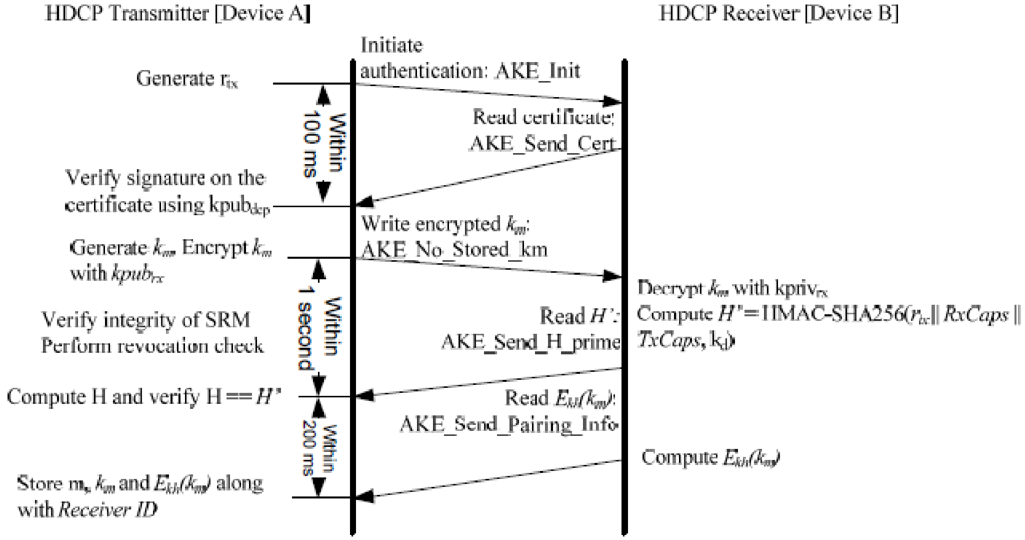


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

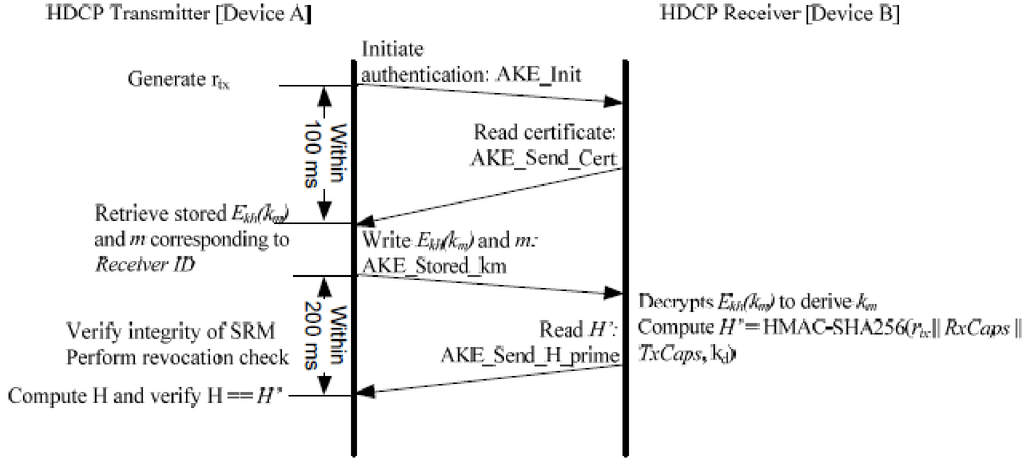
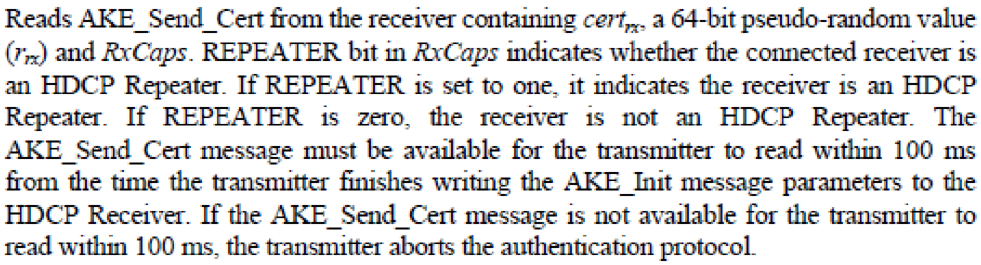


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

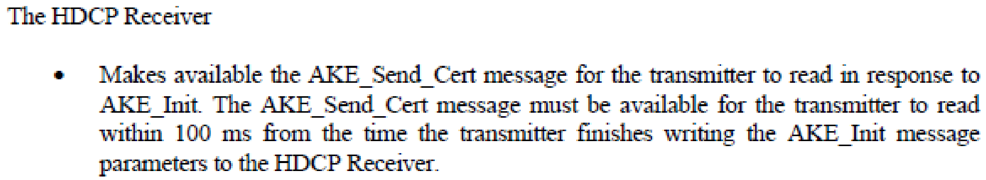
Id.

The Accused Product receives the certificate from the second device as part of the AKE_Send_Cert message.



Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol.

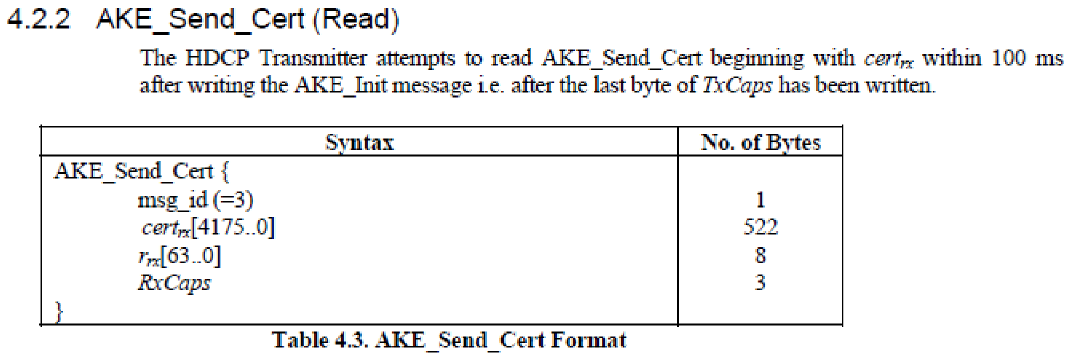
Id. at 13.



The HDCP Receiver

- Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

Id. at 14.



4.2.2 AKE_Send_Cert (Read)

The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

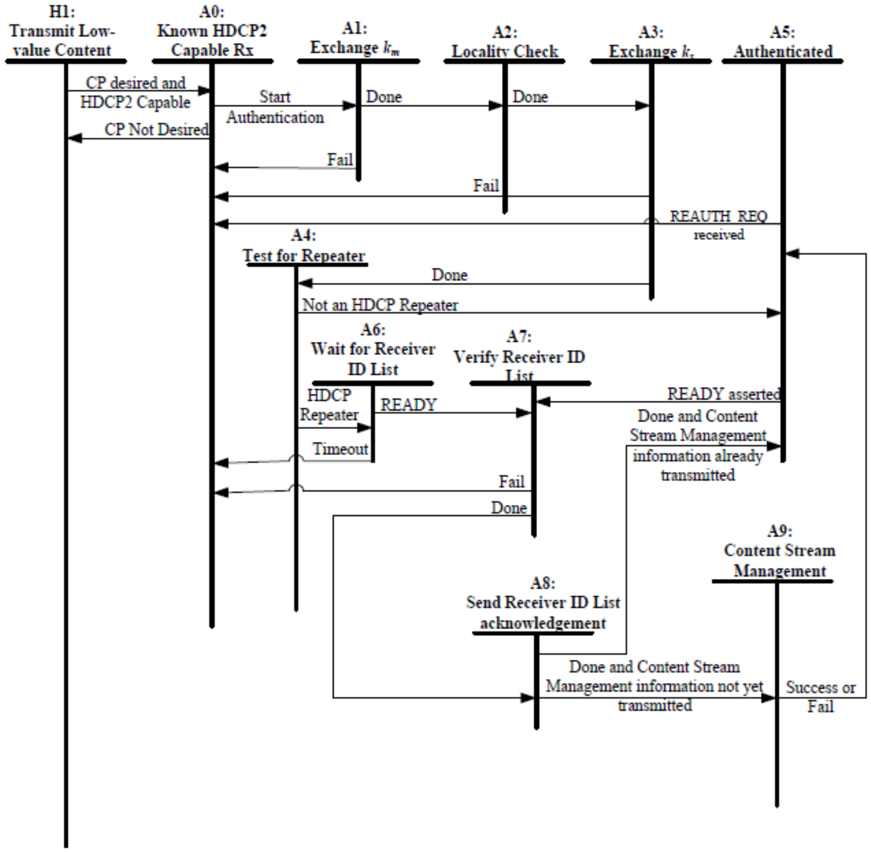
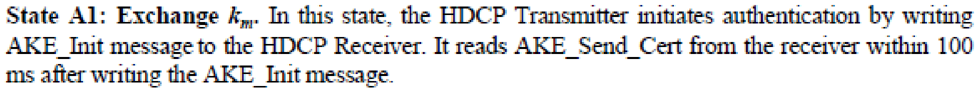
Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

See also:

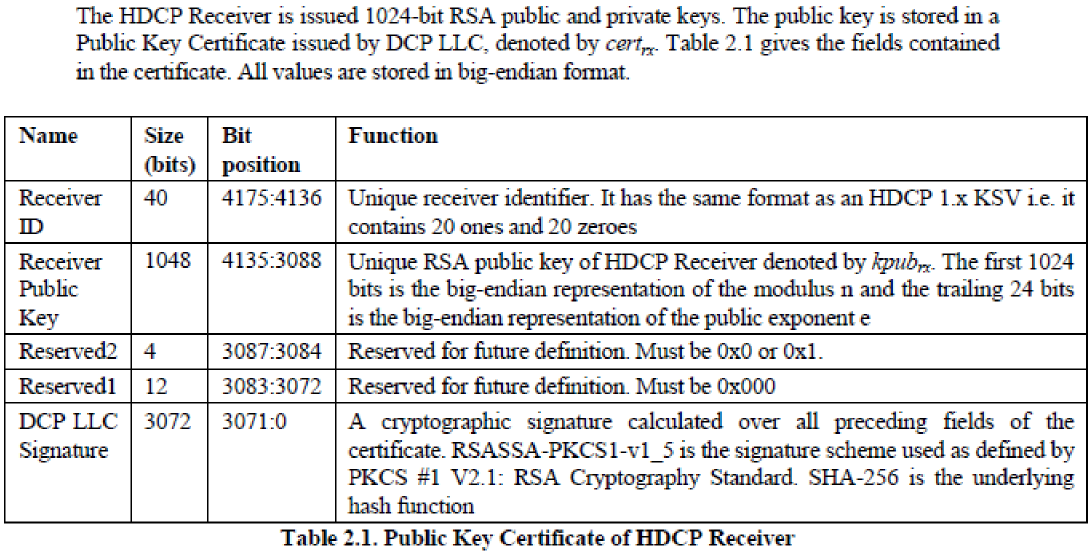
"receive a certificate of the second device, the certificate providing information regarding the second device;"

	 **Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram** *Id.* at 27.  **State A1: Exchange k_m.** In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message. *Id.* at 28.

"determine whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate;"

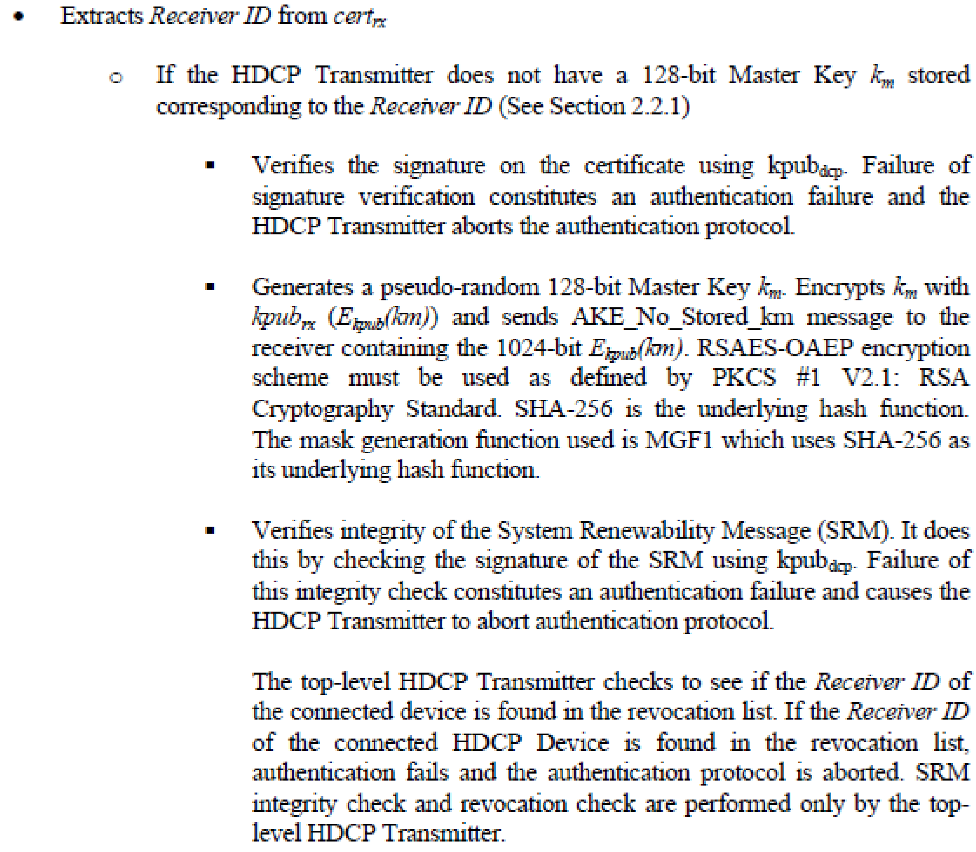
<table>
<tr>
<td>determine whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate;</td>
<td>
The processor of the Accused Product is arranged to determine whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate.

The Accused Product determines, as part of the Authentication and Key Exchange (AKE) stage, whether the second device is compliant with a set of compliance rules using the information provided in the certificate, <i>e.g.</i>, $cert_{rx}$. For example, $cert_{rx}$ includes a Receiver ID, Receiver Public Key, and a cryptographic signature.


The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.

The Accused Product determines, for example, whether the certificate conforms with the format established by the HDCP specification (<i>see id</i>. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.
</td>
</tr>
</table>



- Extracts *Receiver ID* from $cert_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using $kpub_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2, *available at* https://digital-cp.com/sites/default/files/HDCP%20License%20Agreement_March%206%2C%202017_FOR%20REVIEW%20ONLY.pdf.

"determine whether the second device is compliant with a set of compliance rules utilizing said information provided in said certificate;"

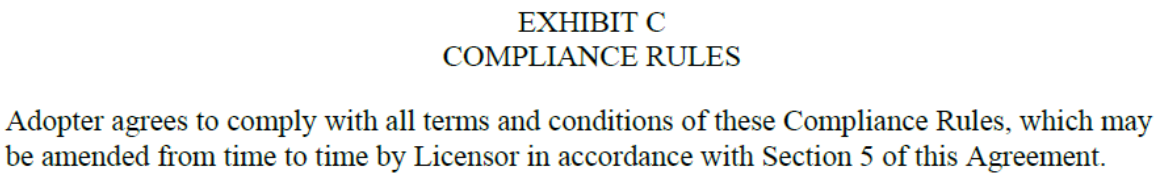


EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

Id. at Exhibit C.

See also:

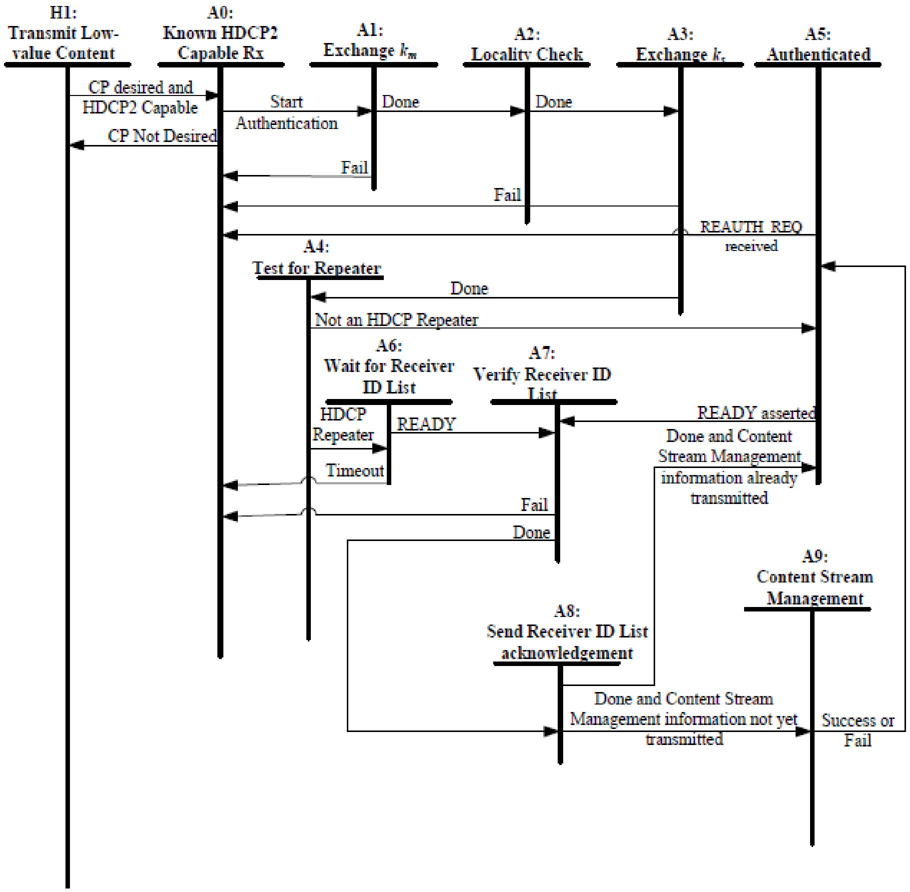
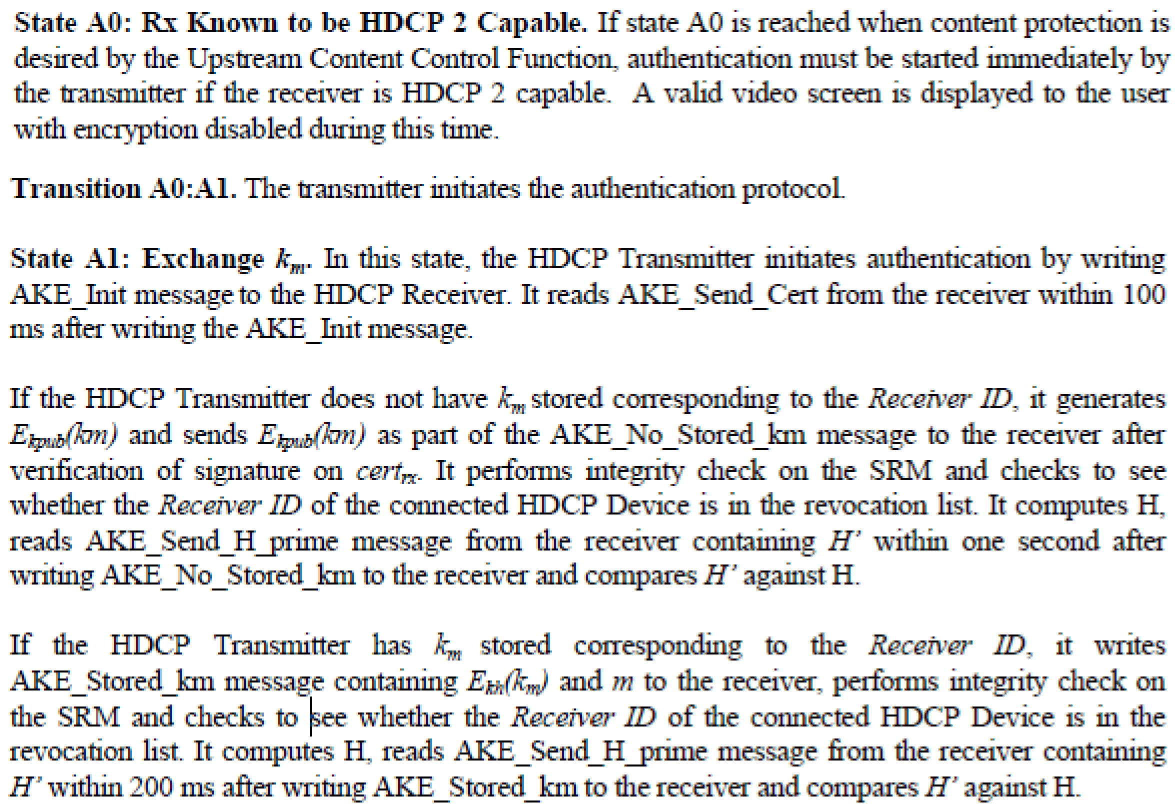
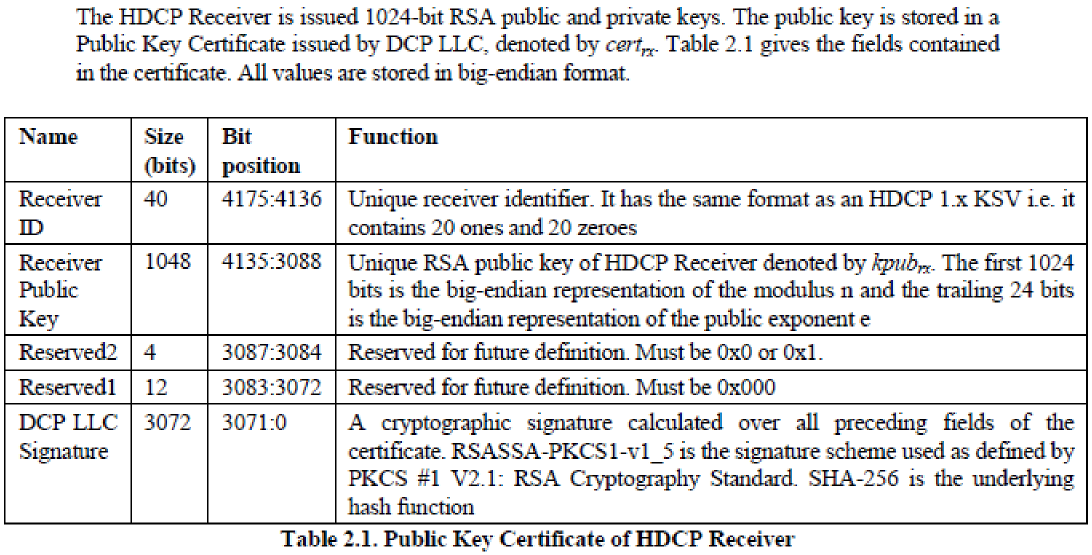
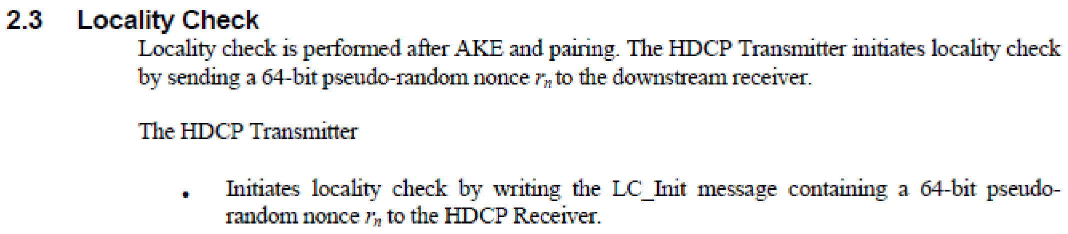


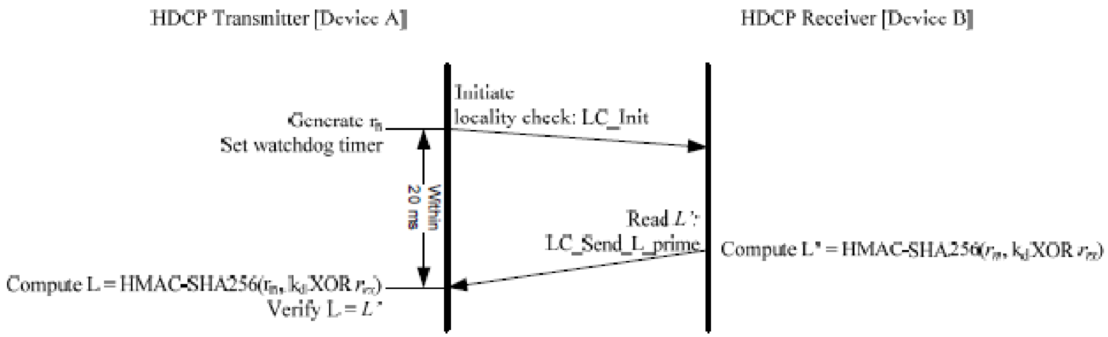
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

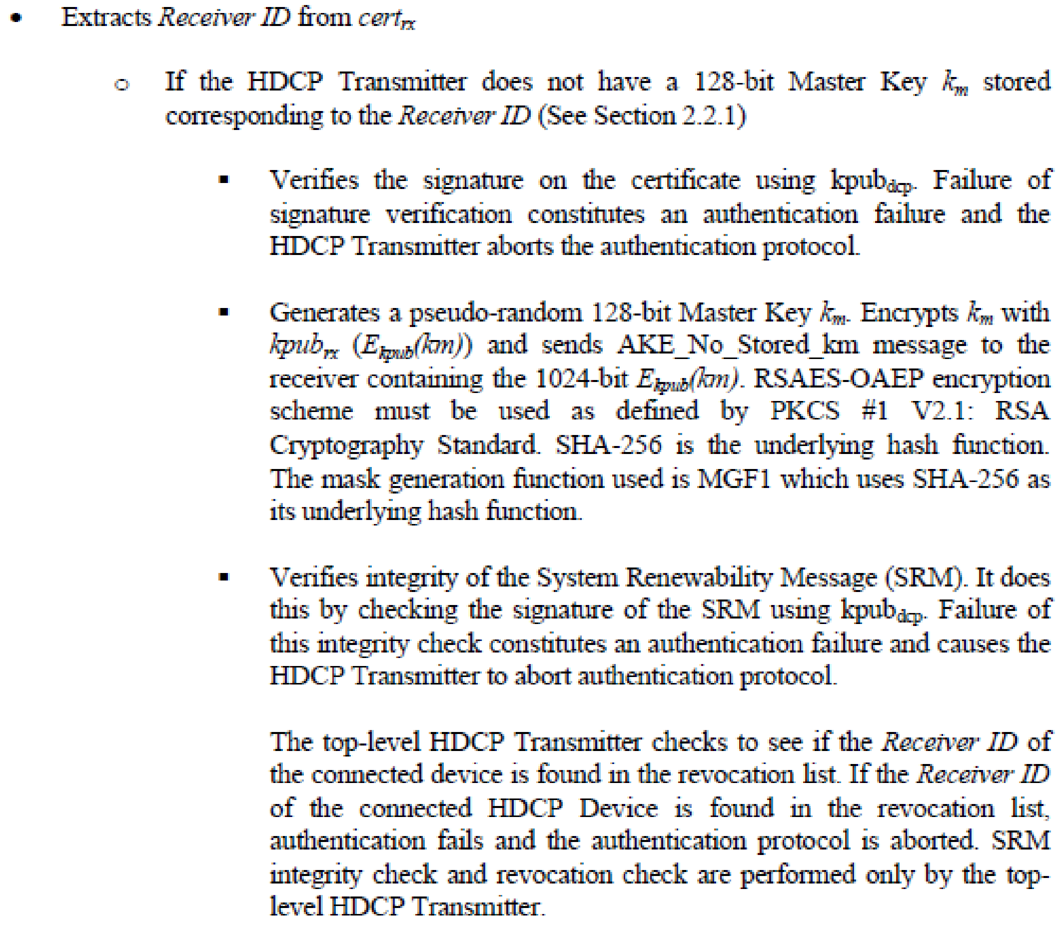
	HDMI HDCP 2.2 at 27.  **State A0: Rx Known to be HDCP 2 Capable.** If state A0 is reached when content protection is desired by the Upstream Content Control Function, authentication must be started immediately by the transmitter if the receiver is HDCP 2 capable. A valid video screen is displayed to the user with encryption disabled during this time. **Transition A0:A1.** The transmitter initiates the authentication protocol. **State A1: Exchange k_m.** In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message. If the HDCP Transmitter does not have k_m stored corresponding to the *Receiver ID*, it generates $E_{kpub}(km)$ and sends $E_{kpub}(km)$ as part of the AKE_No_Stored_km message to the receiver after verification of signature on $cert_{rx}$. It performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within one second after writing AKE_No_Stored_km to the receiver and compares *H'* against H. If the HDCP Transmitter has k_m stored corresponding to the *Receiver ID*, it writes AKE_Stored_km message containing $E_{kh}(k_m)$ and *m* to the receiver, performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within 200 ms after writing AKE_Stored_km to the receiver and compares *H'* against H. *Id.* at 28.

"provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;"

<table>
<tr>
<td>provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;</td>
<td>
<p>The processor of the Accused Product is arranged to provide a first signal, <i>e.g.</i>, the LC_Init message including r_n, to the second device depending when the second device is determined to be compliant with the set of compliance rules.</p>
<p>The Accused Product provides the LC_Init message including r_n to the second device when the Accused Product determines in the Authentication and Key Exchange (AKE) stage that the certificate, $cert_{rx}$, indicates that the second device is compliant with the set of compliance rules. For example, the certificate, $cert_{rx}$, includes a Receiver ID, Receiver Public Key, and a cryptographic signature.</p>

<p>The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.</p>
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
<p>Table 2.1. Public Key Certificate of HDCP Receiver</p>
<p>HDMI HDCP 2.2 at 11.</p>

<p>2.3 Locality Check</p>
<p>Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.</p>
<p>The HDCP Transmitter</p>
Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
<p><i>Id</i>. at 16.</p>
</td>
</tr>
</table>

"provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;"

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** *Id*. at 17. The Accused Product provides the LC_Init message to the second device when, for example, the Accused Product determines that the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.



- Extracts *Receiver ID* from *cert*$_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using kpub$_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with *kpub*$_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using kpub$_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

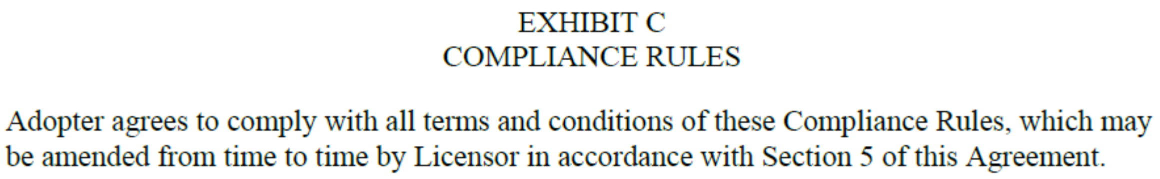
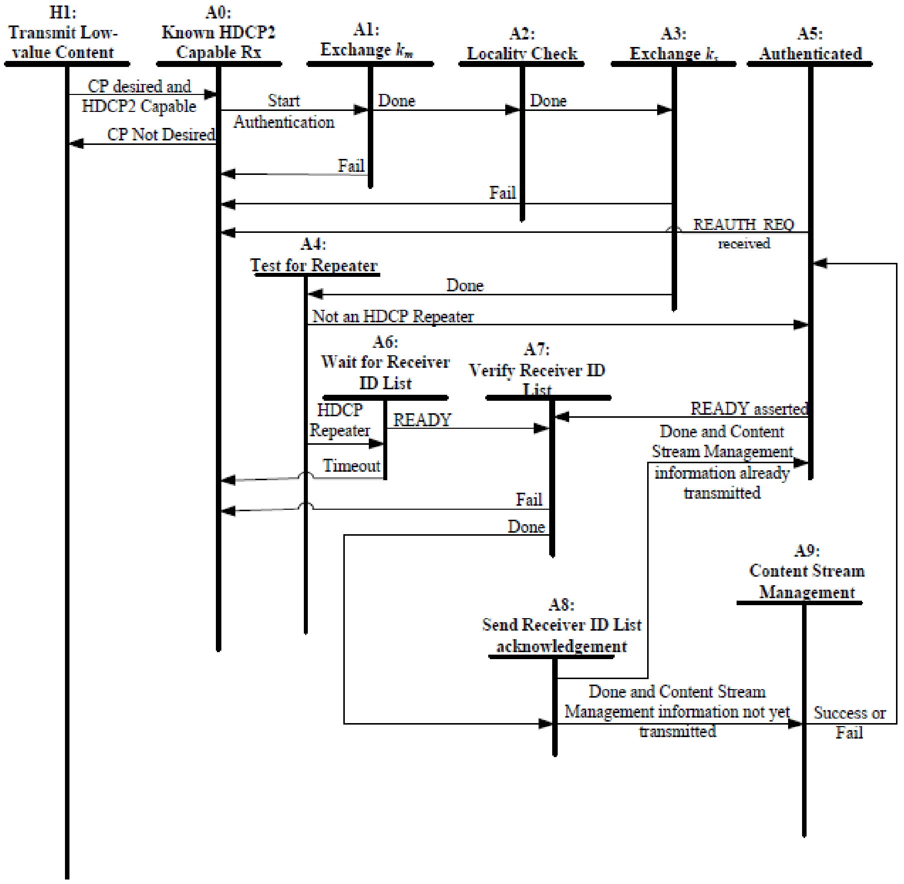
Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

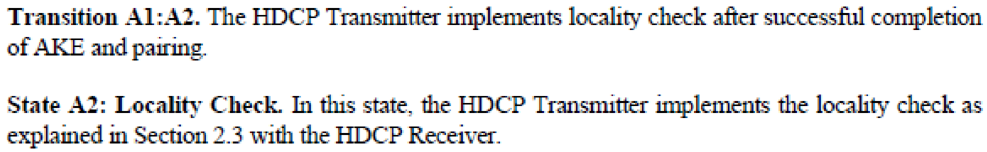
> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2.

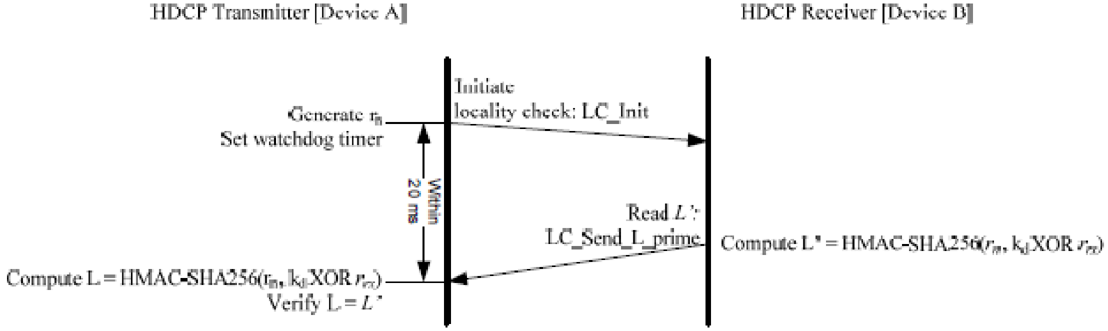
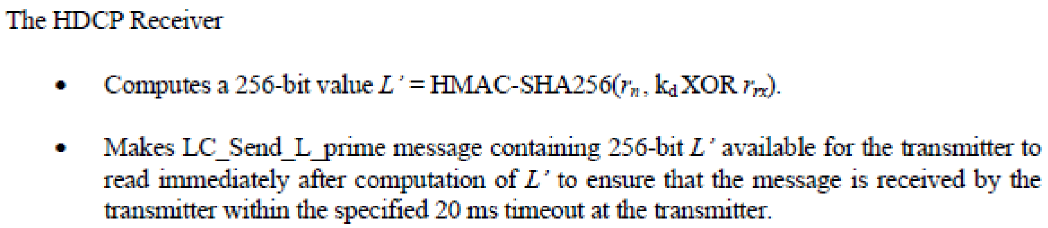
"provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;"

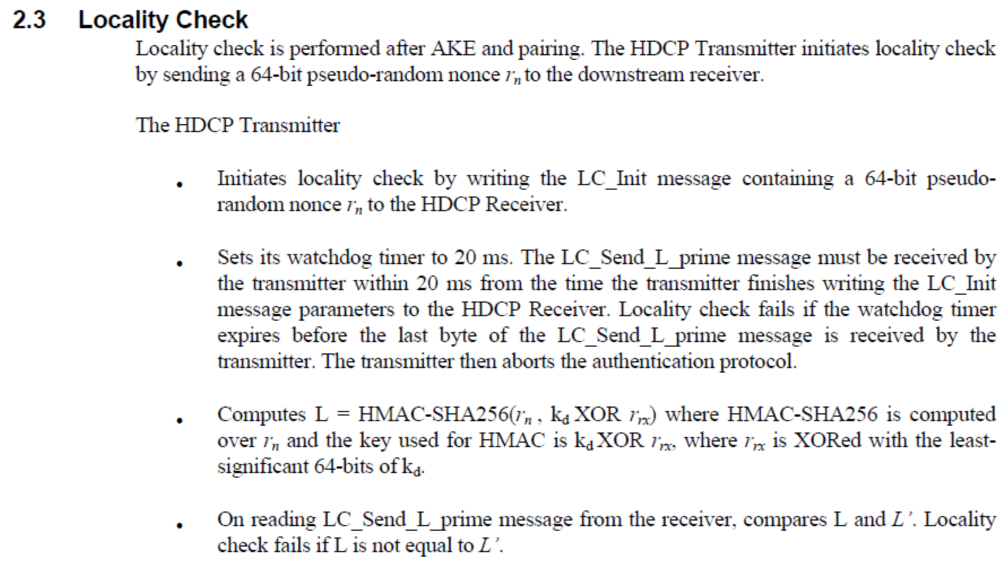
	 EXHIBIT C COMPLIANCE RULES Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement. *Id.* at Exhibit C. *See also*:  **Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram**

"provide a first signal to the second device depending when the second device is determined to be compliant with the set of compliance rules;"

	HDMI HDCP 2.2 at 27.  **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id.* at 28.

"receive a second signal from the second device after providing the first signal;"

receive a second signal from the second device after providing the first signal;	The processor of the Accused Product is arranged to receive a second signal, *e.g.*, the LC_Send_L_prime message including L', from the second device after providing the first signal, *e.g.*, the LC_Init message including r_n.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** HDMI HDCP 2.2 at 17.  The HDCP Receiver • Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$. • Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. *Id.*



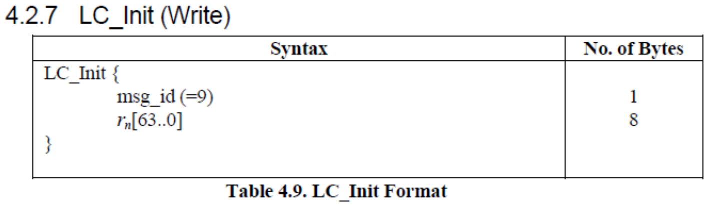
2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes $L = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$ where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

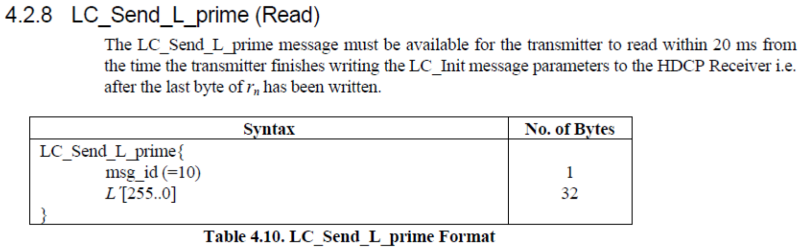


4.2.7 LC_Init (Write)

Syntax	No. of Bytes
LC_Init {	
msg_id (=9)	1
r_n[63..0]	8
}	

Table 4.9. LC_Init Format

Id. at 59.



4.2.8 LC_Send_L_prime (Read)

The LC_Send_L_prime message must be available for the transmitter to read within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver i.e. after the last byte of r_n has been written.

Syntax	No. of Bytes
LC_Send_L_prime{	
msg_id (=10)	1
L'[255..0]	32
}	

Table 4.10. LC_Send_L_prime Format

Id.

"receive a second signal from the second device after providing the first signal;"

See also:

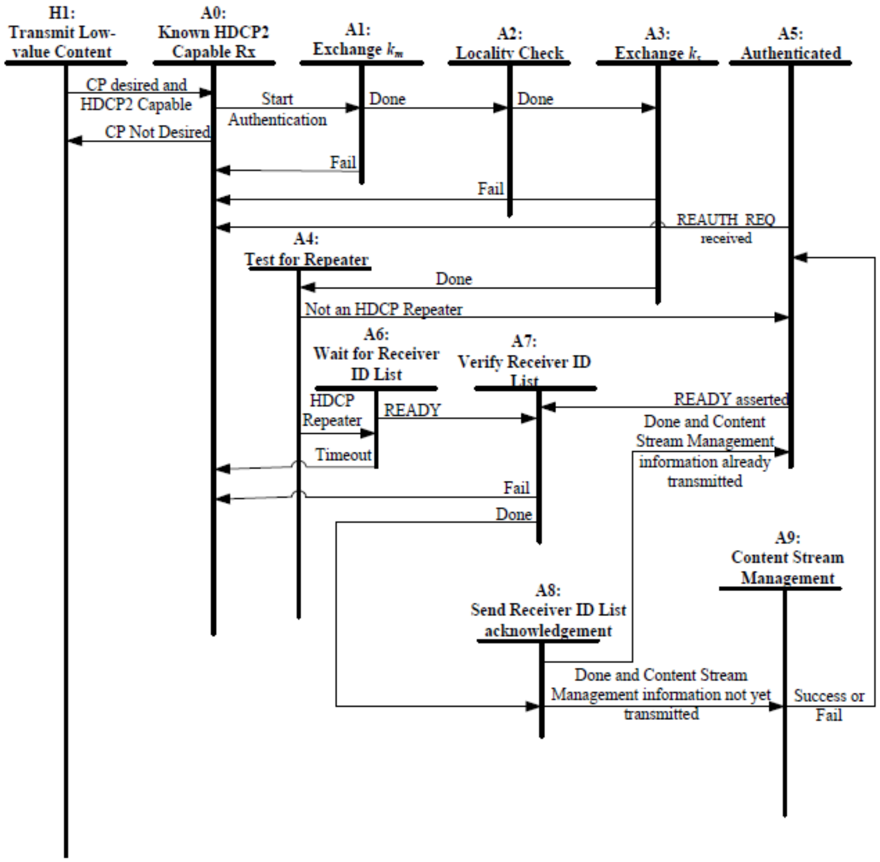


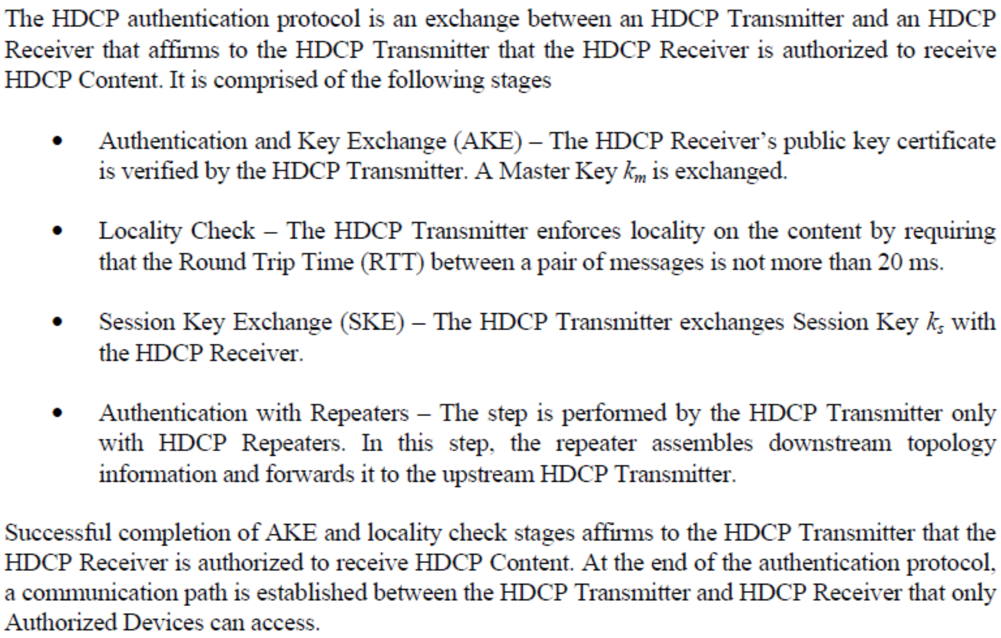
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

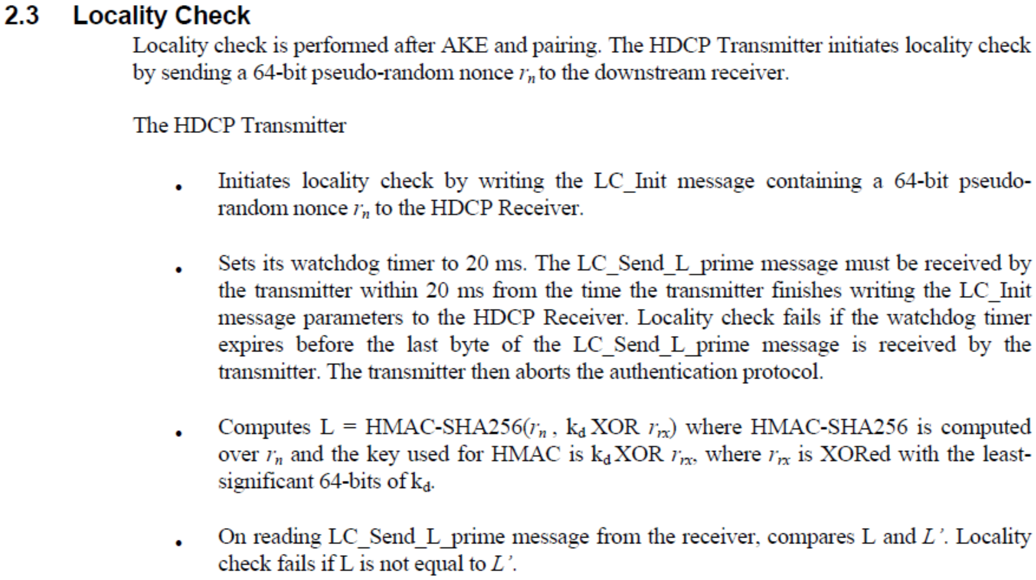
> **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing.
>
> **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"determine whether the second signal is derived from a secret known by the first device;"

determine whether the second signal is derived from a secret known by the first device;	The processor of the Accused Product is arranged to determine whether the second signal, *e.g.*, L', is derived from a secret known by the Accused Product (first device). The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Locality Check requires the Accused Product (transmitter) to determine that L' received via the LC_Send_L_prime message is derived from a secret by matching L' to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m).

"determine whether the second signal is derived from a secret known by the first device;"

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

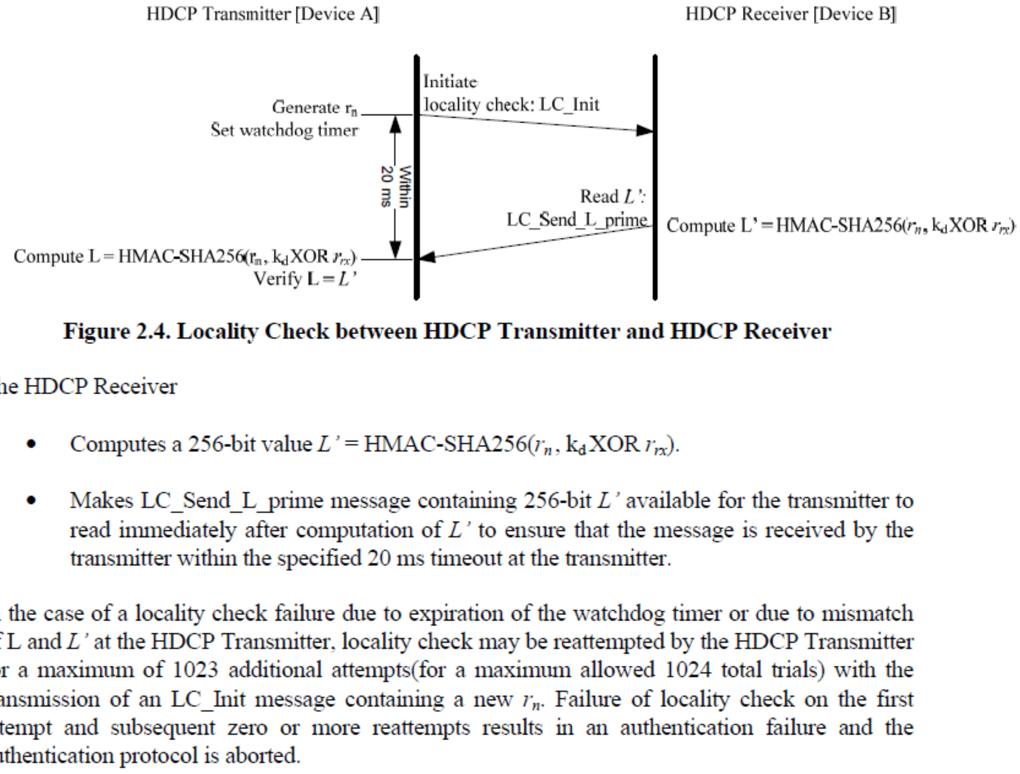


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

The second signal, *e.g.*, L', is derived from a secret.

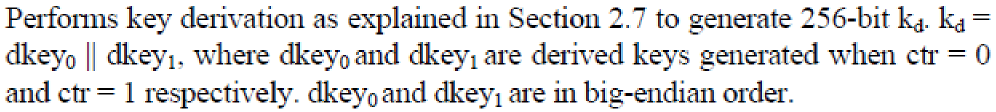
The value of L' is derived from k_d.

Compute $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$

Id.

The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

"determine whether the second signal is derived from a secret known by the first device;"



Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 14-15.

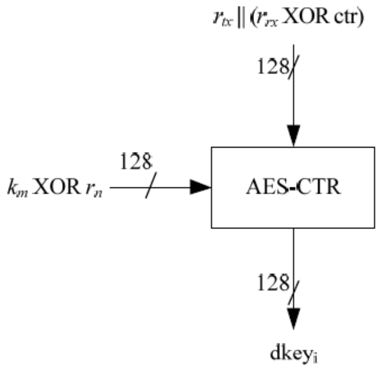
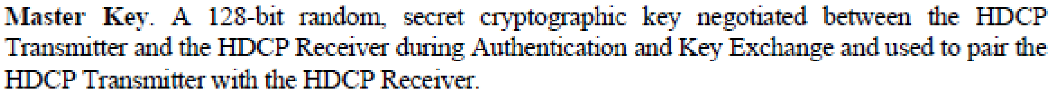


Figure 2.10. Key Derivation

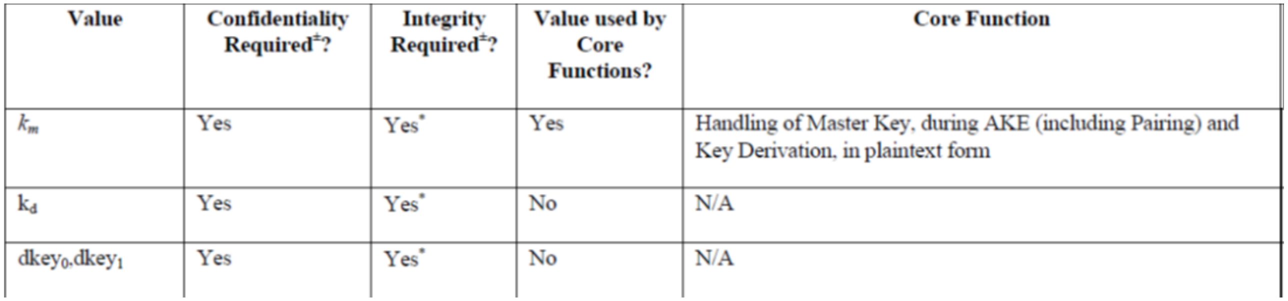
Id. at 25.



Master Key. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver.

Id. at 8.

Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret.



Value	Confidentiality Required$^{\pm}$?	Integrity Required$^{\pm}$?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
$dkey_0$,$dkey_1$	Yes	Yes*	No	N/A

Id. at 67 (abridged).

The Accused Product generates and/or stores the Master Key, k_m, a secret, and thus knows k_m.

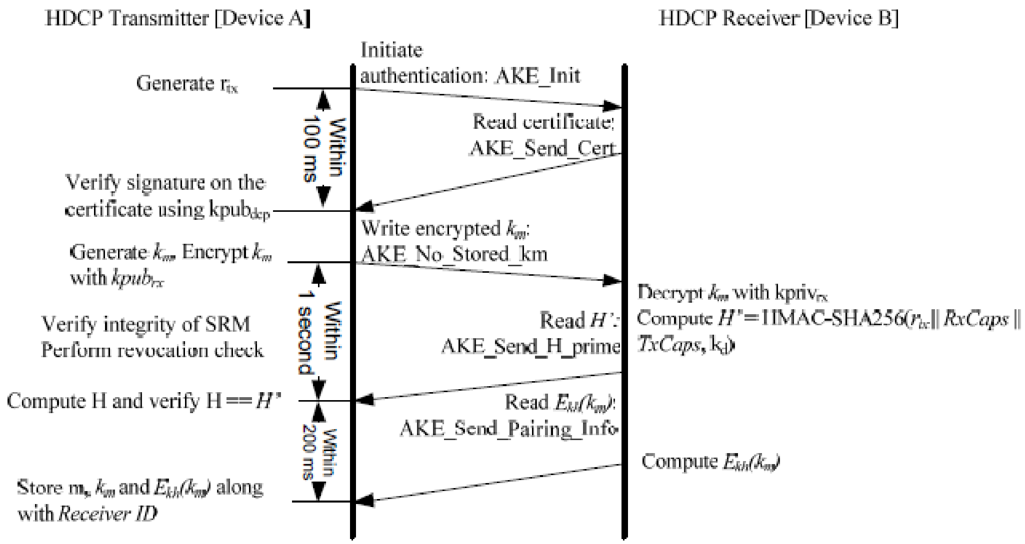
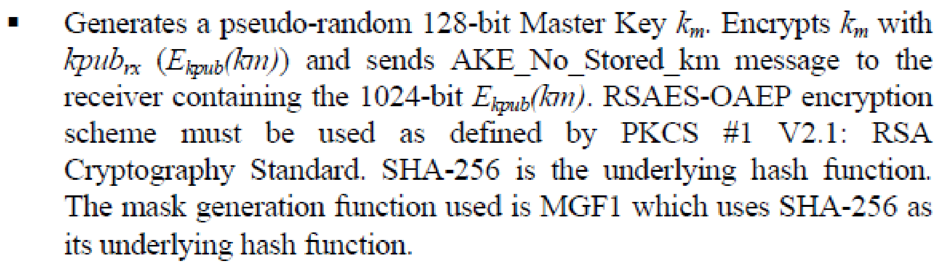


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

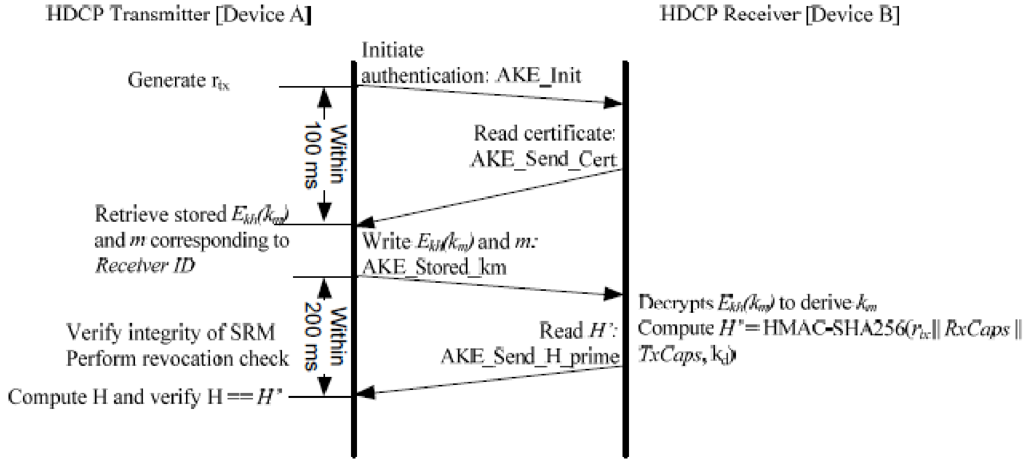
Id. at 12.



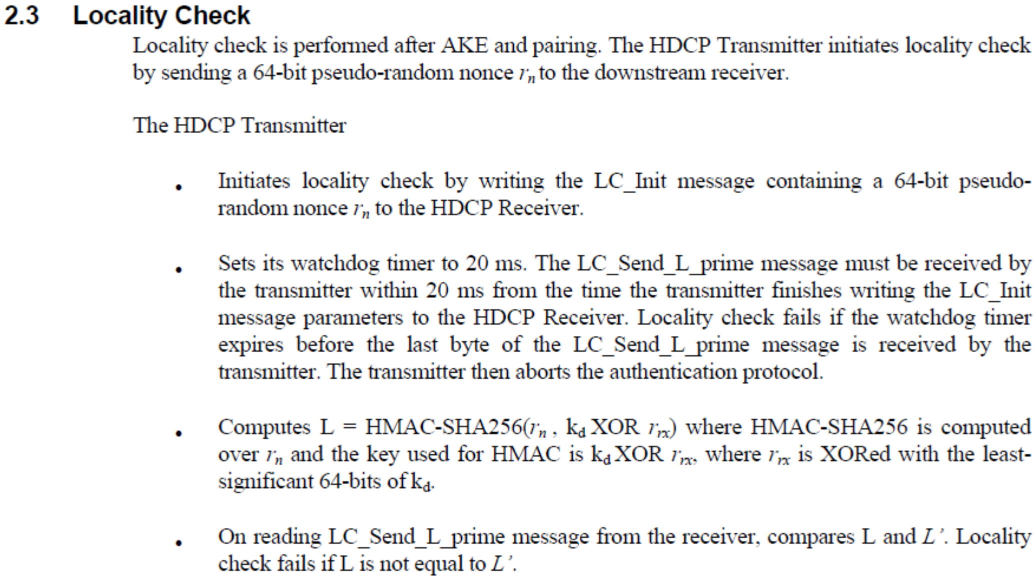
- Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.

Id. at 13.

"determine whether the second signal is derived from a secret known by the first device;"

	 **Figure 2.2. Authentication and Key Exchange (With Stored k_m)** *Id.* at 12. - Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit *m* corresponding to the *Receiver ID* of the HDCP Receiver *Id.* at 14. The Accused Product also knows k_d, which is a secret.

"determine whether the second signal is derived from a secret known by the first device;"



2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

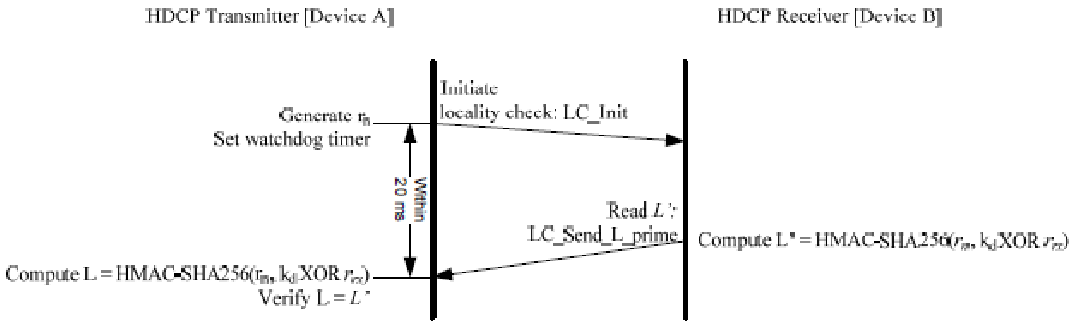


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

Id. at 17.

See also:

"determine whether the second signal is derived from a secret known by the first device;"

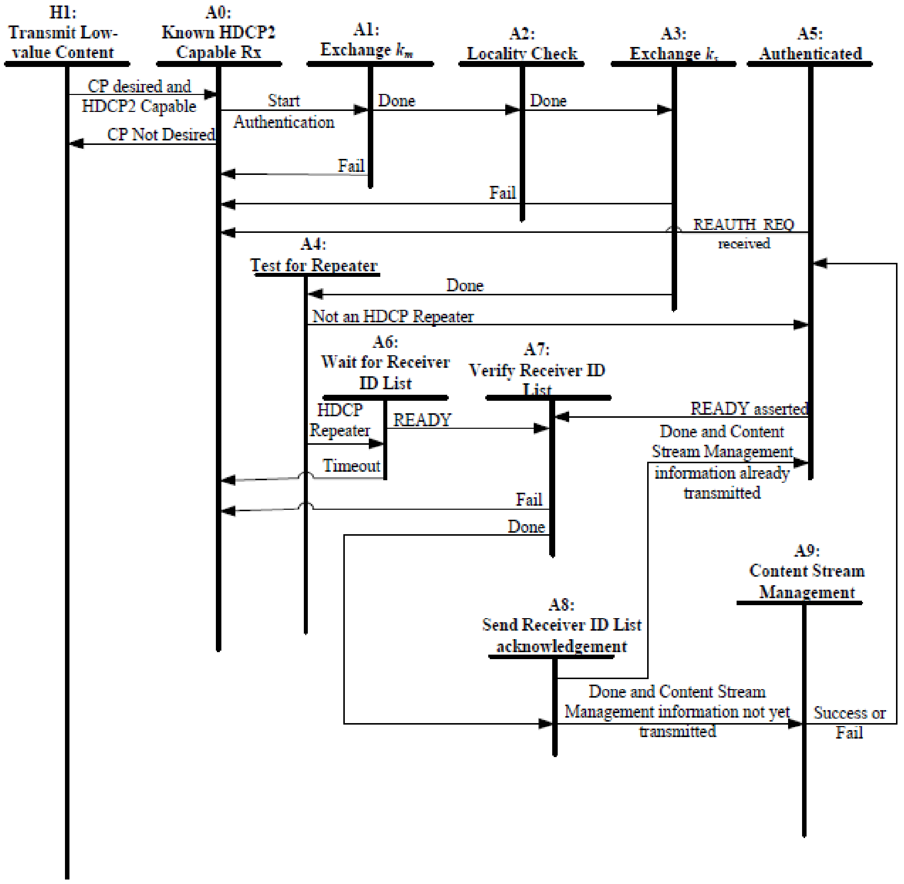


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

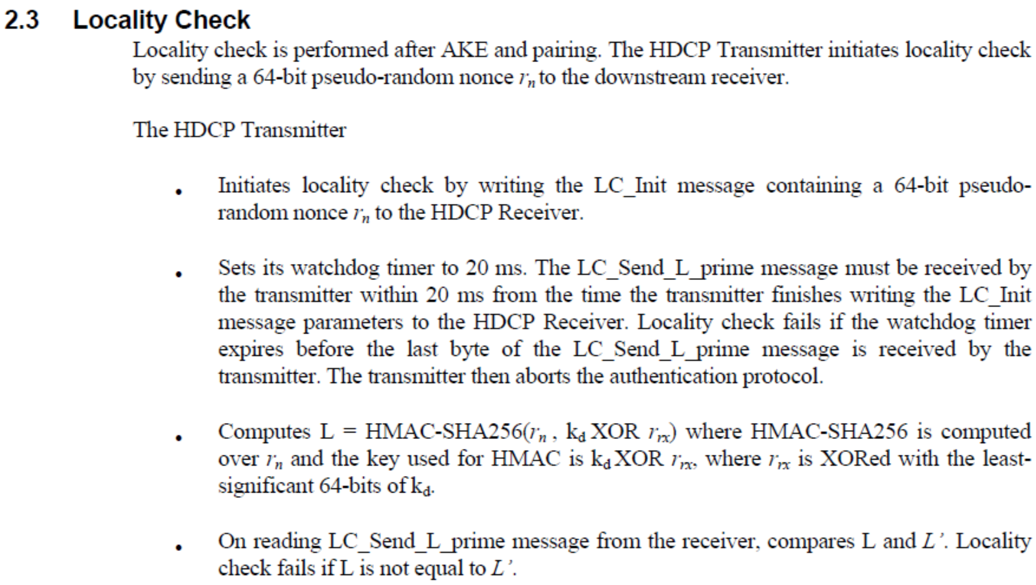
Id. at 27.

Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

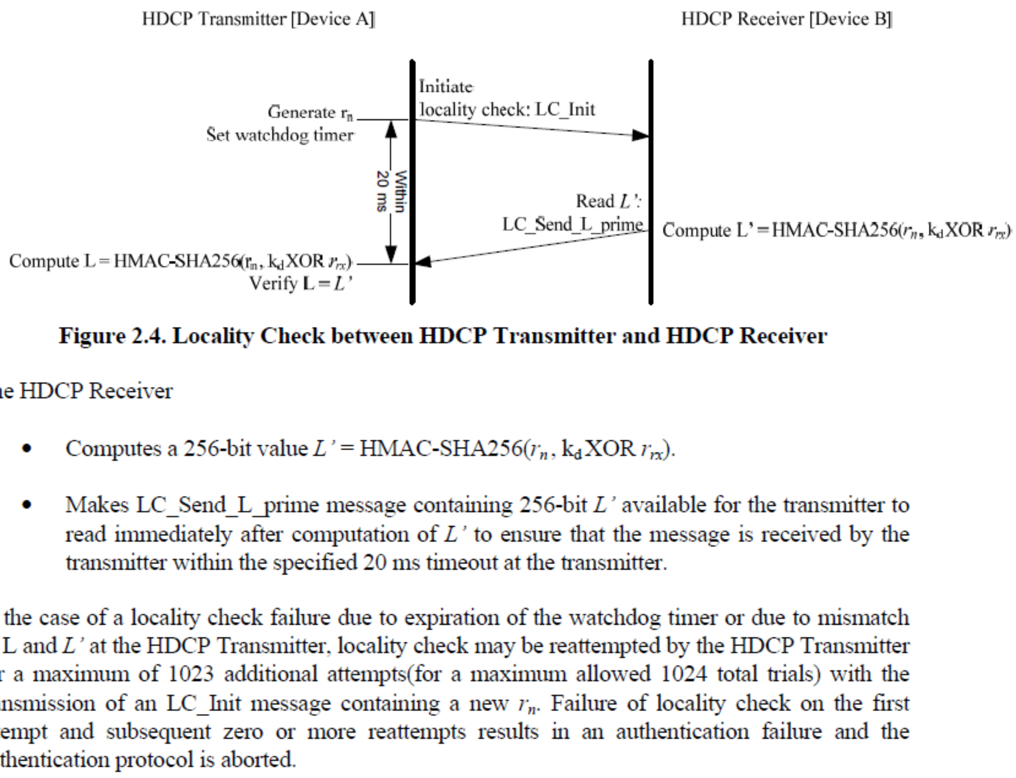
State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and"

determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and	The processor of the Accused Product is arranged to determine whether a time difference between providing the first signal, *e.g.*, the LC_Init message including r_n, and receiving the second signal, *e.g.*, the LC_Send_L_prime message including L', is less than a predetermined time. The Locality Check requires the Accused Product to determine that the time between the providing of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than a predetermined time of 20 ms.  **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n, k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id.* at 16.

"determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and"

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** The HDCP Receiver - Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$. - Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted. *Id*. at 17. *See also*:

"determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and"

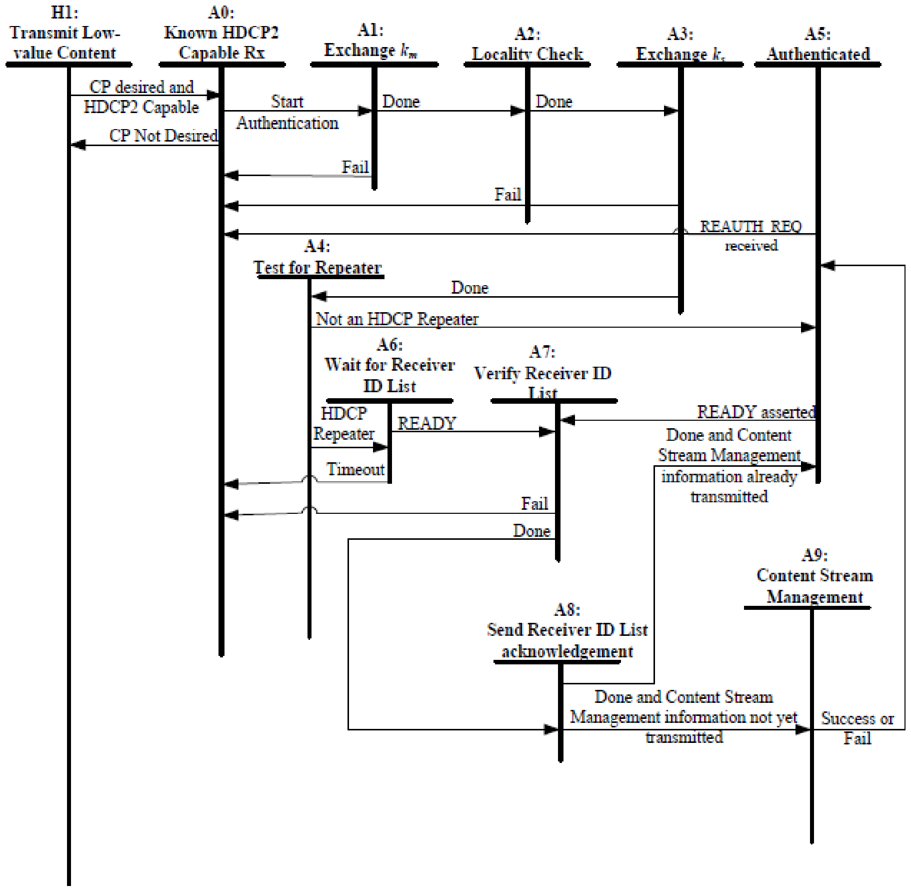


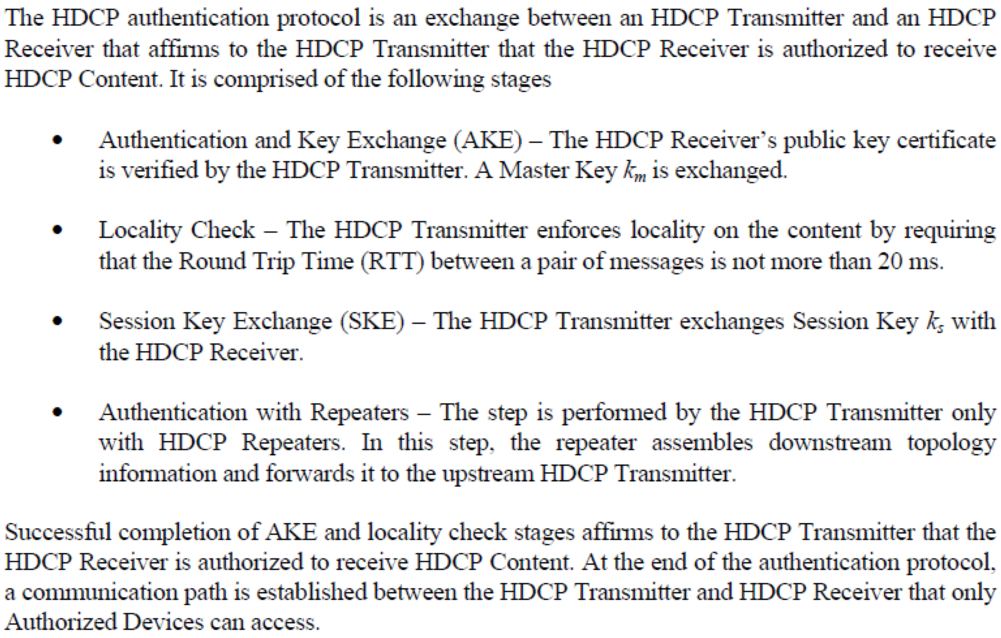
Figure 2.12. **HDCP Transmitter Authentication Protocol State Diagram**

Id. at 27.

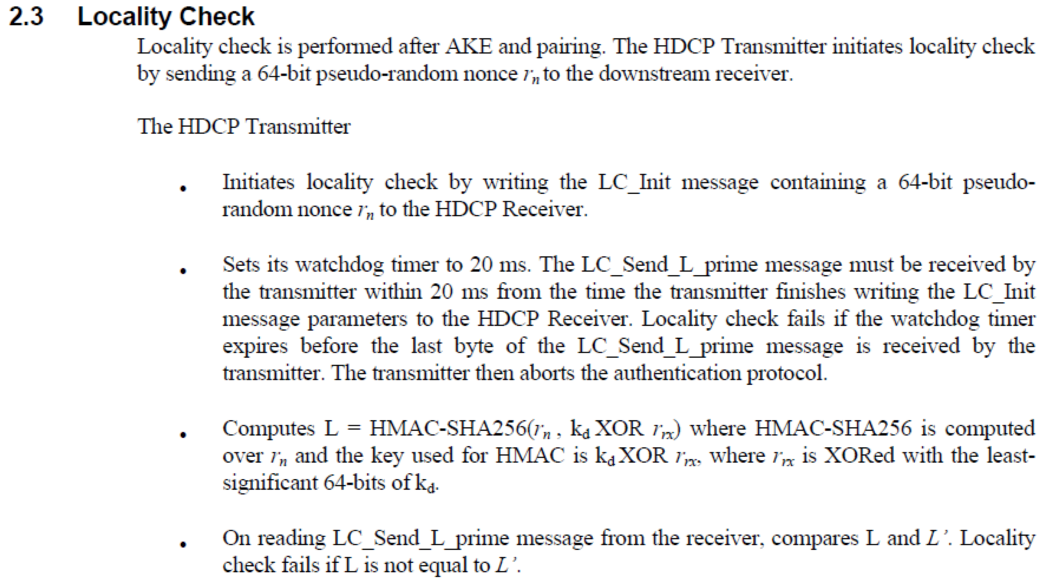
> **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing.
>
> **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.	The processor of the Accused Product is arranged to allow the protected content to be provided to the second device when at least the second signal, *e.g.*, L', is determined to be derived from the secret and the time difference is less than the predetermined time. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Accused Product provides protected content to the second device when, as part of the Locality Check: the L' received via the LC_Send_L_prime message is derived from a secret (as determined by matching L' to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m)); and a time between the providing of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than a predetermined time of 20 ms.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

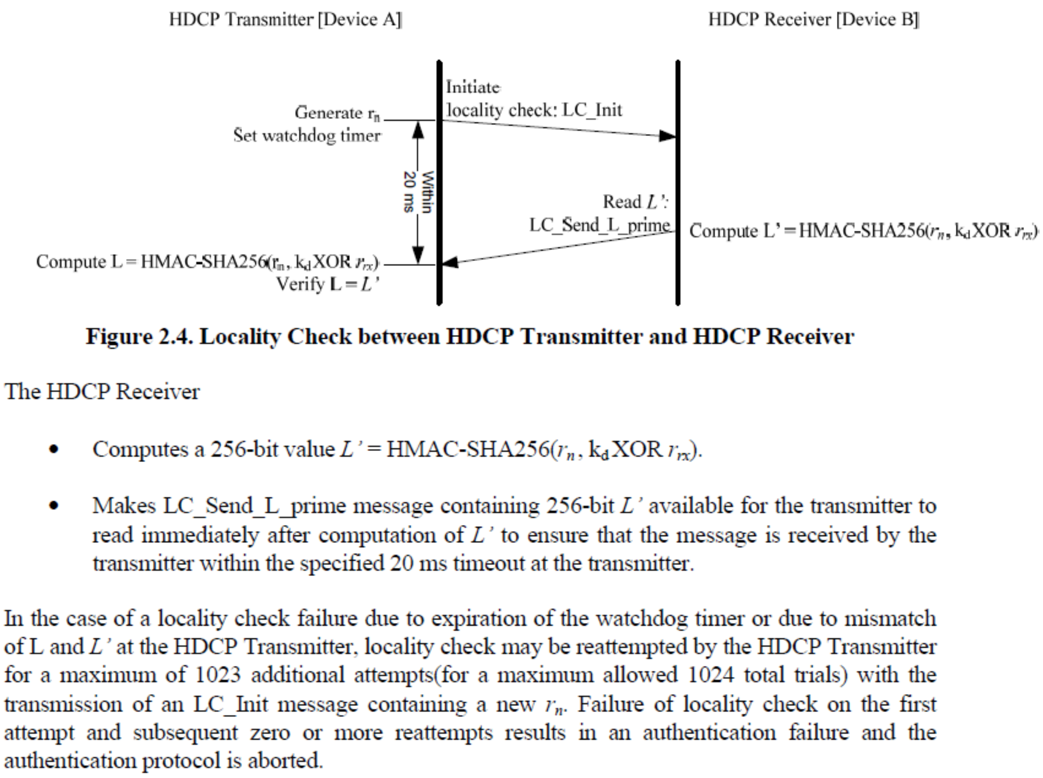


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value $L' =$ HMAC-SHA256(r_n, k_d XOR r_{rx}).
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

The second signal, *e.g.*, L', is derived from a secret.

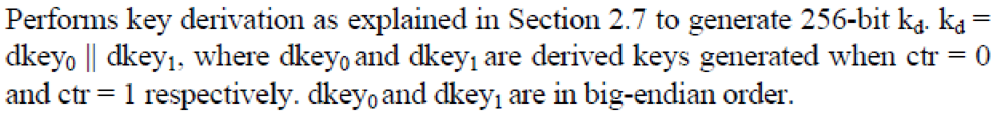
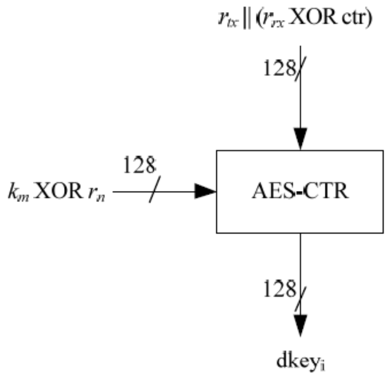
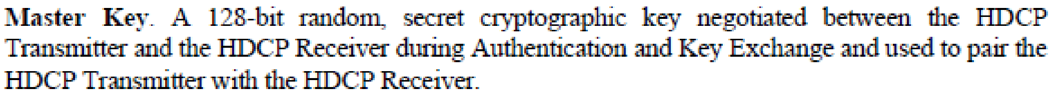
The value of L' is derived from k_d.

Compute L' = HMAC-SHA256(r_n, k_d XOR r_{rx})

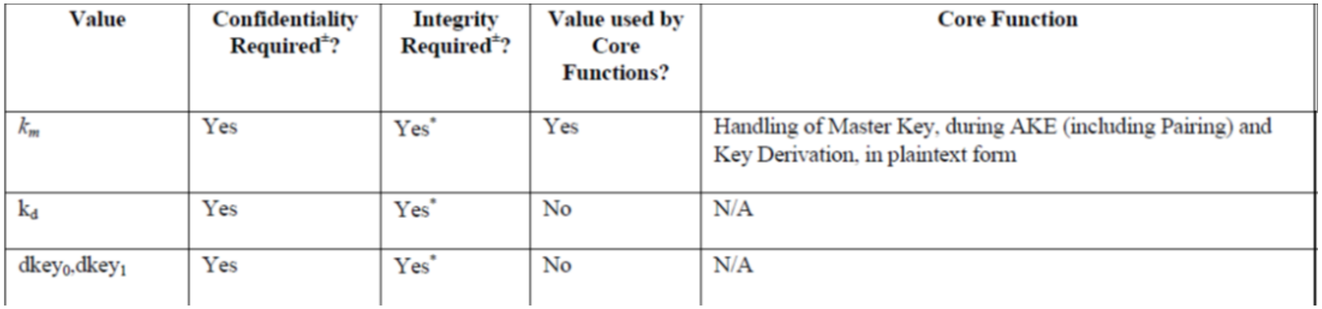
Id.

The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

	 Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order. *Id.* at 14-15.  **Figure 2.10. Key Derivation** *Id.* at 25.  **Master Key**. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver. *Id.* at 8. Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret.

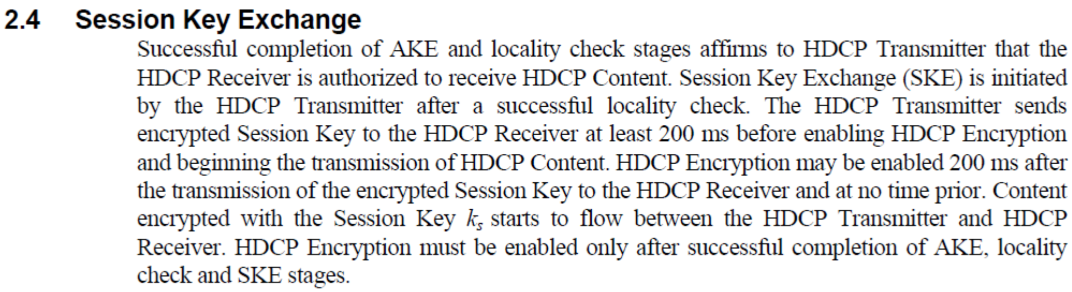
"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."



Value	Confidentiality Required$^{\pm}$?	Integrity Required$^{\pm}$?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
dkey$_0$,dkey$_1$	Yes	Yes*	No	N/A

Id. at 67 (abridged).

The Accused Product proceeds to session key exchange and providing of the protected content to the second device after successful completion of the AKE stage and Locality Check.

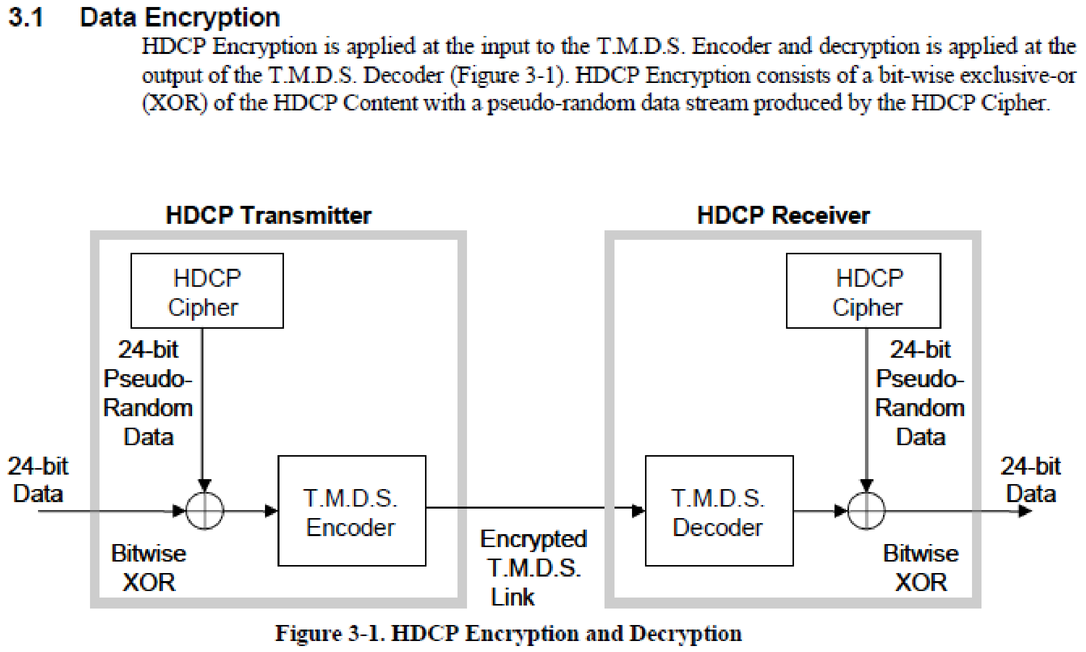


2.4 Session Key Exchange

Successful completion of AKE and locality check stages affirms to HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. Session Key Exchange (SKE) is initiated by the HDCP Transmitter after a successful locality check. The HDCP Transmitter sends encrypted Session Key to the HDCP Receiver at least 200 ms before enabling HDCP Encryption and beginning the transmission of HDCP Content. HDCP Encryption may be enabled 200 ms after the transmission of the encrypted Session Key to the HDCP Receiver and at no time prior. Content encrypted with the Session Key k_s starts to flow between the HDCP Transmitter and HDCP Receiver. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages.

Id. at 17.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

	 **3.1 Data Encryption** HDCP Encryption is applied at the input to the T.M.D.S. Encoder and decryption is applied at the output of the T.M.D.S. Decoder (Figure 3-1). HDCP Encryption consists of a bit-wise exclusive-or (XOR) of the HDCP Content with a pseudo-random data stream produced by the HDCP Cipher. **Figure 3-1. HDCP Encryption and Decryption** *Id.* at 50. *See also*:

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

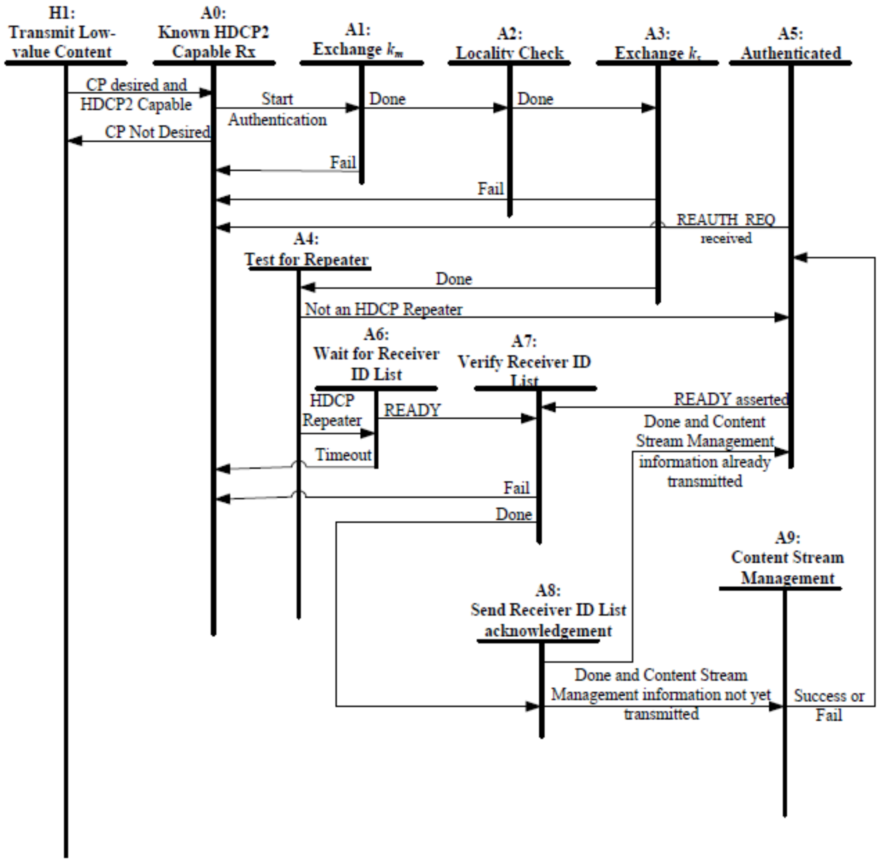
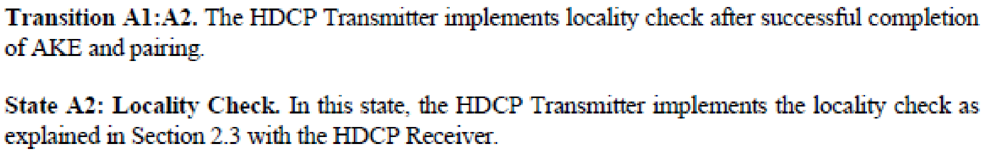


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

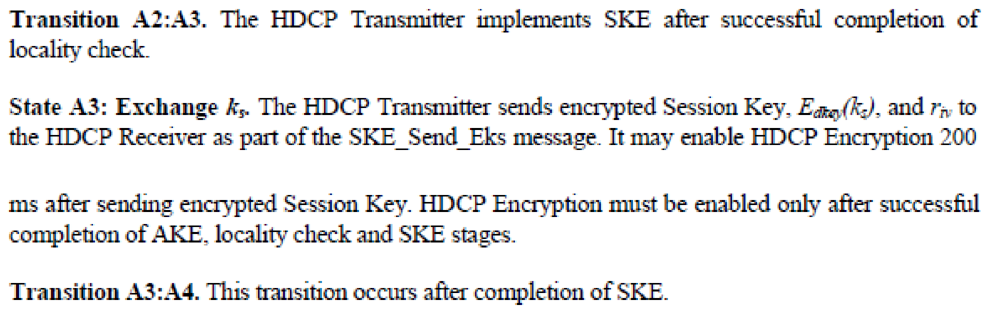


Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"allow the protected content to be provided to the second device when at least the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

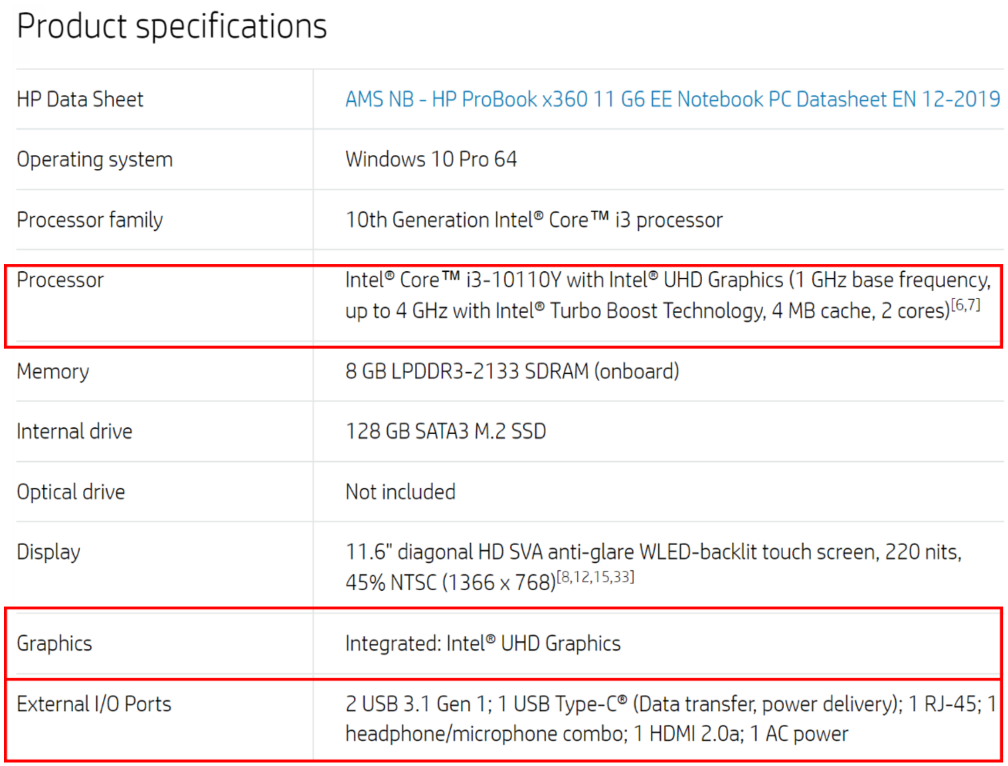
	 **Transition A2:A3.** The HDCP Transmitter implements SKE after successful completion of locality check. **State A3: Exchange k_s.** The HDCP Transmitter sends encrypted Session Key, $E_{dkey}(k_s)$, and r_{rv} to the HDCP Receiver as part of the SKE_Send_Eks message. It may enable HDCP Encryption 200 ms after sending encrypted Session Key. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages. **Transition A3:A4.** This transition occurs after completion of SKE. *Id*. at 28-29.

U.S. Patent No. 9,436,809: Claim 17

"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"

17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:	Each of the HP Product and the Intel Product is a system for controlling the transmission of protected content from a content provider to a requesting device, and is referred to herein as an "Accused Product." For example, the HP Product is an HDMI transmitter with HDCP 2.2 for controlling transmission of protected content to a requesting device, such as an HDMI receiver with HDCP 2.2.  HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. The HP Product includes an HDMI 2.0a port and a 10th Generation Intel® Core™ i3-10110Y Processor (the "Intel Processor") integrated with the Intel UHD Graphics 615 graphics processor (the "Intel GPU") that enable delivery of protected content to another device.

"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"



Product specifications

HP Data Sheet	AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019
Operating system	Windows 10 Pro 64
Processor family	10th Generation Intel® Core™ i3 processor
Processor	Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]
Memory	8 GB LPDDR3-2133 SDRAM (onboard)
Internal drive	128 GB SATA3 M.2 SSD
Optical drive	Not included
Display	11.6" diagonal HD SVA anti-glare WLED-backlit touch screen, 220 nits, 45% NTSC (1366 x 768)[8,12,15,33]
Graphics	Integrated: Intel® UHD Graphics
External I/O Ports	2 USB 3.1 Gen 1; 1 USB Type-C® (Data transfer, power delivery); 1 RJ-45; 1 headphone/microphone combo; 1 HDMI 2.0a; 1 AC power

Id. *See also* NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.

The Intel Processor supports HDCP 2.2 via HDMI 2.0a.

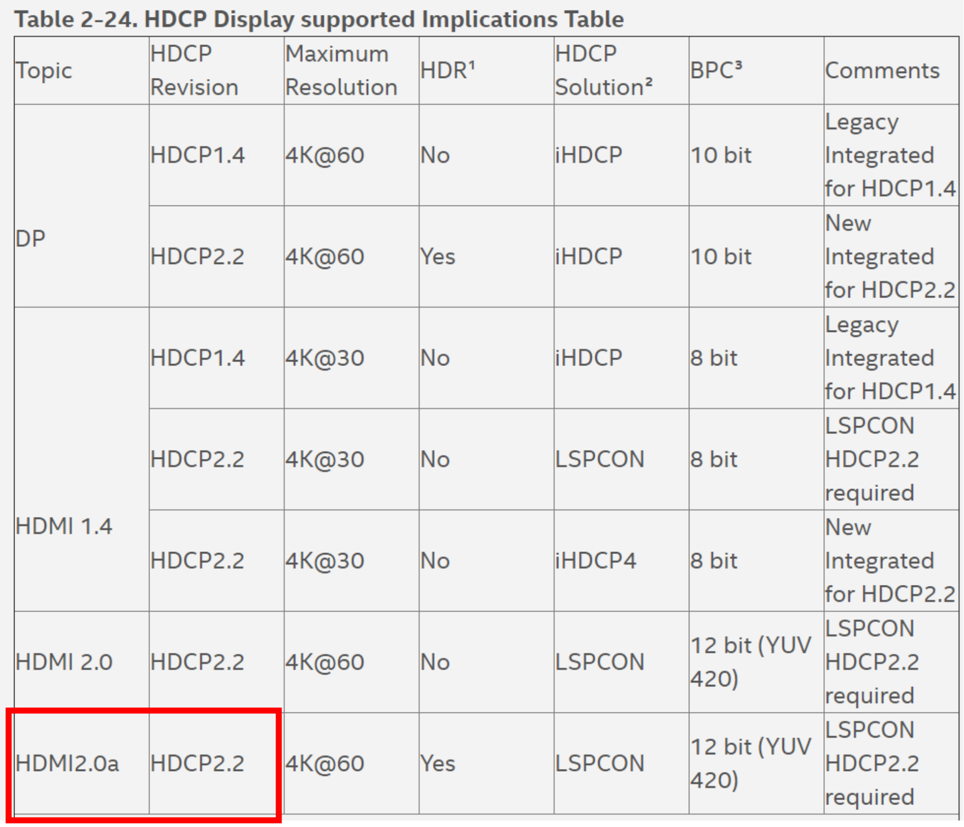


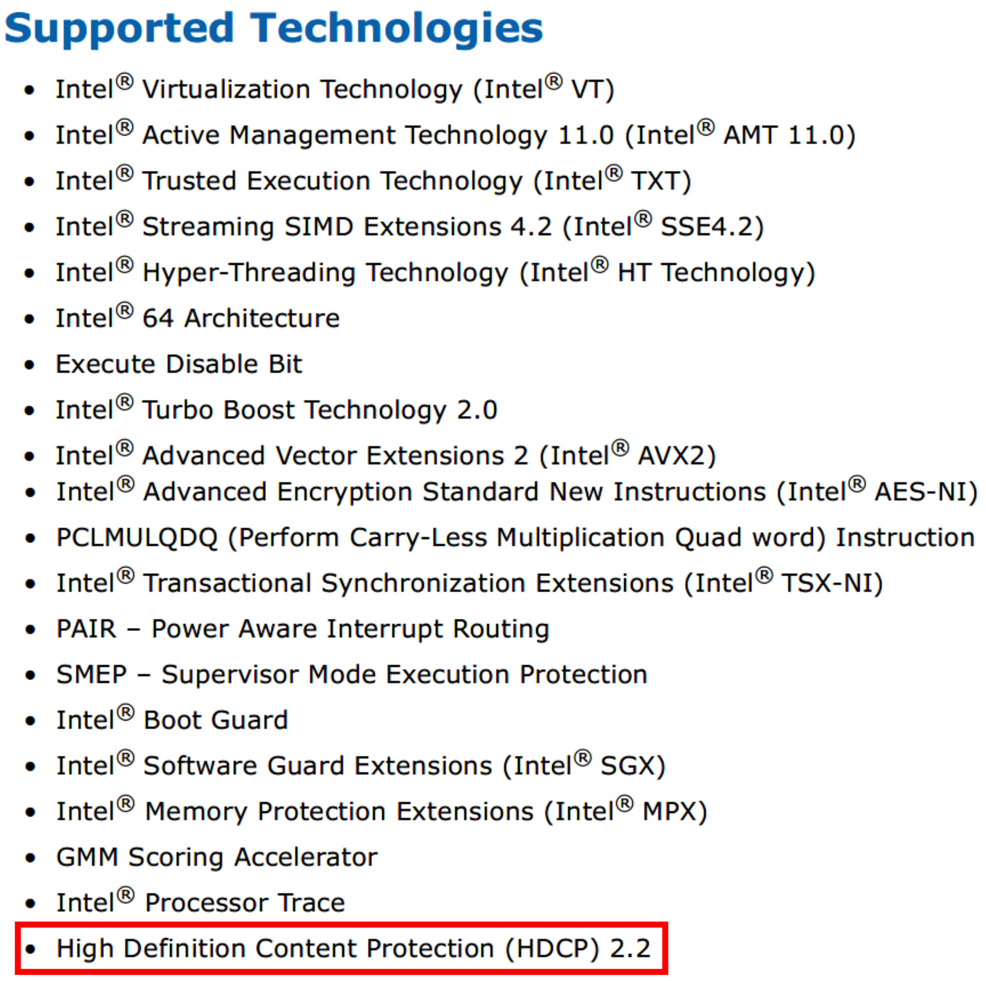
Table 2-24. HDCP Display supported Implications Table

Topic	HDCP Revision	Maximum Resolution	HDR[1]	HDCP Solution[2]	BPC[3]	Comments
DP	HDCP1.4	4K@60	No	iHDCP	10 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@60	Yes	iHDCP	10 bit	New Integrated for HDCP2.2
HDMI 1.4	HDCP1.4	4K@30	No	iHDCP	8 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@30	No	LSPCON	8 bit	LSPCON HDCP2.2 required
	HDCP2.2	4K@30	No	iHDCP4	8 bit	New Integrated for HDCP2.2
HDMI 2.0	HDCP2.2	4K@60	No	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required
HDMI2.0a	HDCP2.2	4K@60	Yes	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required

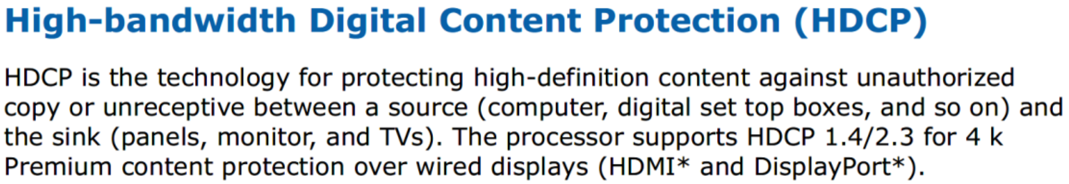
Intel, How to enable High Dynamic Range?, https://www.intel.com/content/www/us/en/support/articles/000032112/graphics/graphics-for-7th-generation-intel-processors.html.

While the above datasheet indicates that Intel Core processors have supported HDCP 2.2 over HDMI2.0a as of the 7th Generation, Intel's documentation for its current, 10th-generation Core processors indicates that support for HDCP 2.2 is native rather than necessitating LSPCON support.

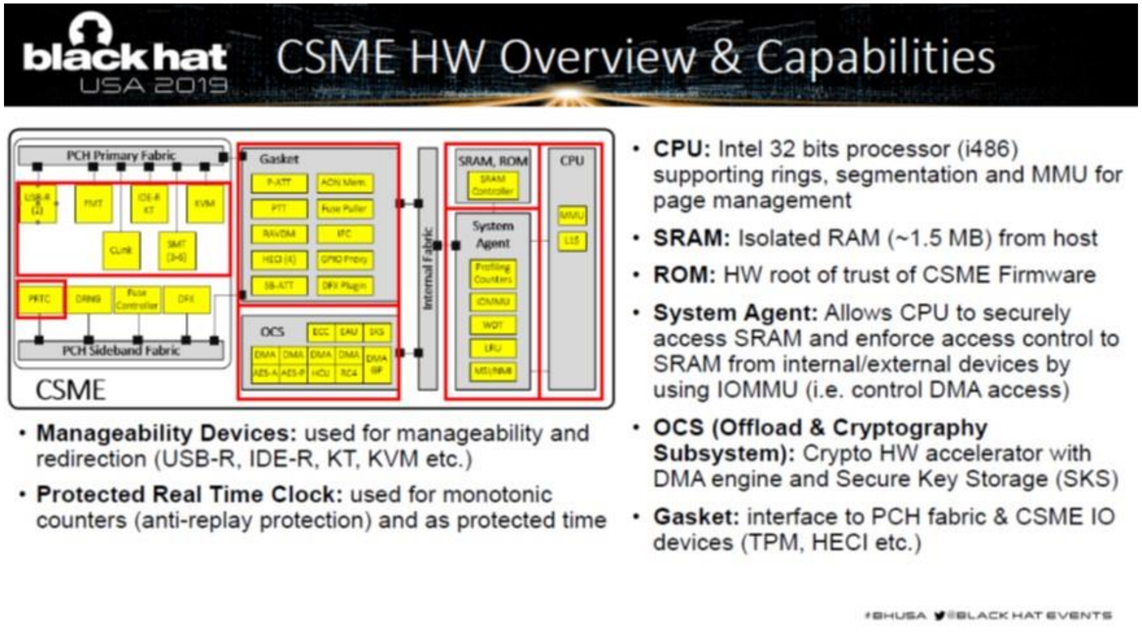
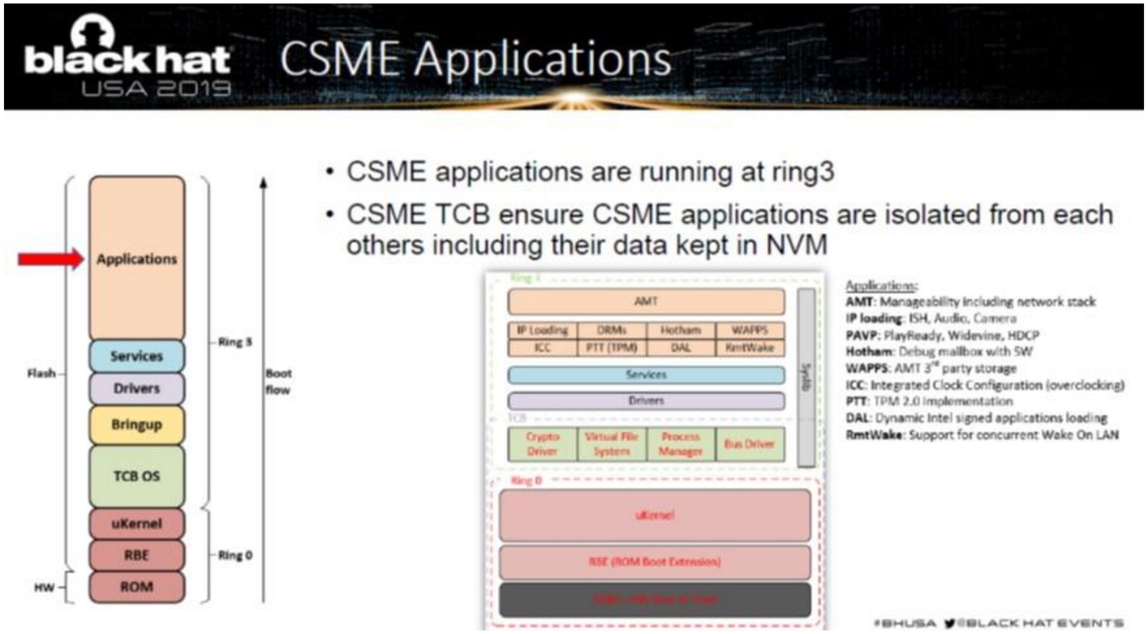
"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"

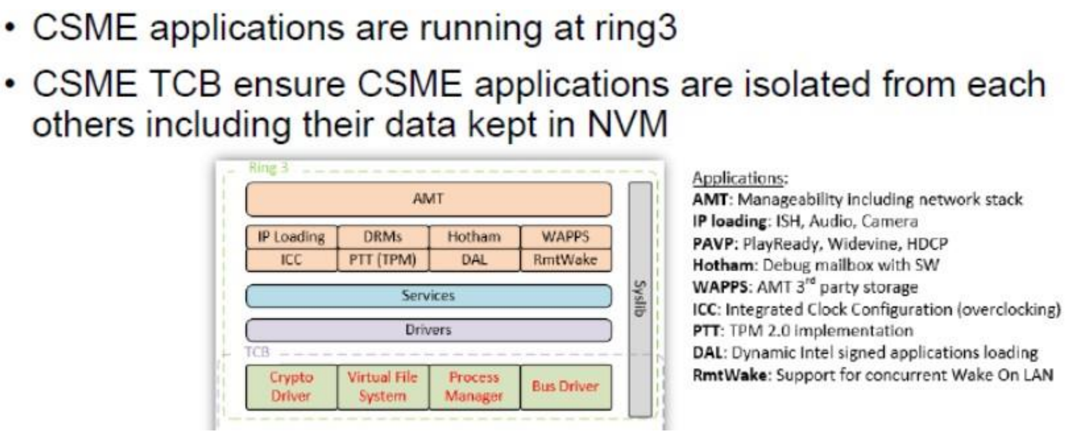
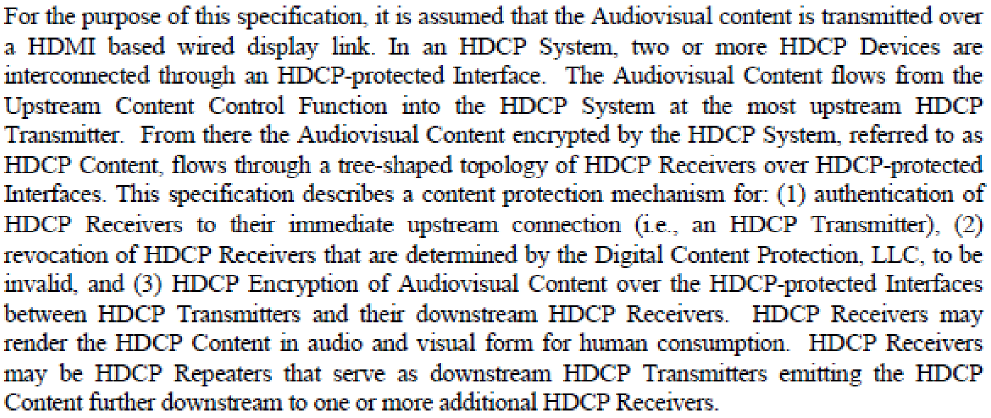
	 **Supported Technologies** - Intel® Virtualization Technology (Intel® VT) - Intel® Active Management Technology 11.0 (Intel® AMT 11.0) - Intel® Trusted Execution Technology (Intel® TXT) - Intel® Streaming SIMD Extensions 4.2 (Intel® SSE4.2) - Intel® Hyper-Threading Technology (Intel® HT Technology) - Intel® 64 Architecture - Execute Disable Bit - Intel® Turbo Boost Technology 2.0 - Intel® Advanced Vector Extensions 2 (Intel® AVX2) - Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI) - PCLMULQDQ (Perform Carry-Less Multiplication Quad word) Instruction - Intel® Transactional Synchronization Extensions (Intel® TSX-NI) - PAIR – Power Aware Interrupt Routing - SMEP – Supervisor Mode Execution Protection - Intel® Boot Guard - Intel® Software Guard Extensions (Intel® SGX) - Intel® Memory Protection Extensions (Intel® MPX) - GMM Scoring Accelerator - Intel® Processor Trace - High Definition Content Protection (HDCP) 2.2 Intel, 10th Generation Intel Core Processors, Datasheet, Volume 1 or 2 (Jul. 2020, rev. 5), *available at* https://cdrdv2.intel.com/v1/dl/getContent/615211, at 11-12. "HDCP is the technology for protecting high-definition content against unauthorized copy … between a source … and the sink …. The [Intel] [P]rocessor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*)."

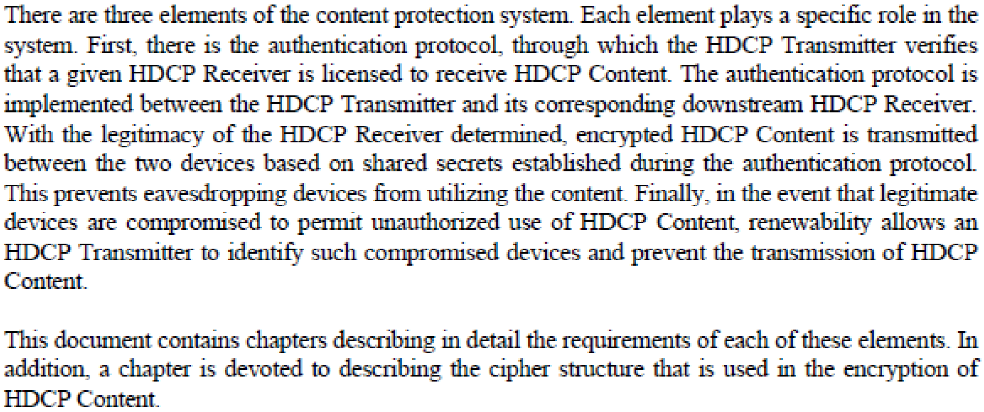
"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"

	 **High-bandwidth Digital Content Protection (HDCP)** HDCP is the technology for protecting high-definition content against unauthorized copy or unreceptive between a source (computer, digital set top boxes, and so on) and the sink (panels, monitor, and TVs). The processor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*). *Id.* at 44 Intel's "UHD" processor nomenclature also indicates support for HDCP 2.2: Another change from 7 Gen to 8 Gen will be in the graphics. Intel is upgrading the nomenclature of the integrated graphics from HD 620 to UHD 620, indicating that the silicon is suited for 4K playback and processing. During our pre-briefing it was categorically stated several times that there was no change between the two, however we have since confirmed that the new chips will come with HDCP 2.2 support as standard for DP1.2a, removing the need for an external LSPCON for this feature. Other than this display controller change however, it appears that these new UHD iGPUs are architecturally the same as their HD predecessors. https://www.anandtech.com/show/11738/intel-launches-8th-generation-cpus-starting-with-kaby-lake-refresh-for-15w-mobile. HDCP 2.2 is implemented in Intel-based systems with Core-i series Processors within the Converged Security & Manageability Engine (CSME) also known as the Management Engine (ME). The CSME contains a processor (x86 core) which executes instructions including but not limited to the uKernel/OS, drivers, services, and applications for the CSME.

"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"

	 **CSME HW Overview & Capabilities** - **Manageability Devices:** used for manageability and redirection (USB-R, IDE-R, KT, KVM etc.) - **Protected Real Time Clock:** used for monotonic counters (anti-replay protection) and as protected time - **CPU:** Intel 32 bits processor (i486) supporting rings, segmentation and MMU for page management - **SRAM:** Isolated RAM (~1.5 MB) from host - **ROM:** HW root of trust of CSME Firmware - **System Agent:** Allows CPU to securely access SRAM and enforce access control to SRAM from internal/external devices by using IOMMU (i.e. control DMA access) - **OCS (Offload & Cryptography Subsystem):** Crypto HW accelerator with DMA engine and Secure Key Storage (SKS) - **Gasket:** interface to PCH fabric & CSME IO devices (TPM, HECI etc.) Behind the Scenes of Intel Security and Manageability Engine, blackhat USA 2019 ("CSME") at 7.  **CSME Applications** - CSME applications are running at ring3 - CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM Applications: **AMT**: Manageability including network stack **IP loading**: ISH, Audio, Camera **PAVP:** PlayReady, Widevine, HDCP **Hotham**: Debug mailbox with SW **WAPPS**: AMT 3rd party storage **ICC**: Integrated Clock Configuration (overclocking) **PTT**: TPM 2.0 implementation **DAL**: Dynamic Intel signed applications loading **RmtWake**: Support for concurrent Wake On LAN

	Id. at 23. One such application is "PAVP" which provides HDCP capabilities within the Intel processor.  - CSME applications are running at ring3 - CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM Applications: **AMT**: Manageability including network stack **IP loading**: ISH, Audio, Camera **PAVP:** PlayReady, Widevine, HDCP **Hotham**: Debug mailbox with SW **WAPPS**: AMT 3rd party storage **ICC**: Integrated Clock Configuration (overclocking) **PTT**: TPM 2.0 implementation **DAL**: Dynamic Intel signed applications loading **RmtWake**: Support for concurrent Wake On LAN *Id*. Upon information and belief, the Accused Product is compliant with the High-bandwidth Digital Content Protection System Revision 2.2 ("HDCP 2.2") protocol. The Accused Product supports HDCP 2.2 for protecting content between devices.  For the purpose of this specification, it is assumed that the Audiovisual content is transmitted over a HDMI based wired display link. In an HDCP System, two or more HDCP Devices are interconnected through an HDCP-protected Interface. The Audiovisual Content flows from the Upstream Content Control Function into the HDCP System at the most upstream HDCP Transmitter. From there the Audiovisual Content encrypted by the HDCP System, referred to as HDCP Content, flows through a tree-shaped topology of HDCP Receivers over HDCP-protected Interfaces. This specification describes a content protection mechanism for: (1) authentication of HDCP Receivers to their immediate upstream connection (i.e., an HDCP Transmitter), (2) revocation of HDCP Receivers that are determined by the Digital Content Protection, LLC, to be invalid, and (3) HDCP Encryption of Audiovisual Content over the HDCP-protected Interfaces between HDCP Transmitters and their downstream HDCP Receivers. HDCP Receivers may render the HDCP Content in audio and visual form for human consumption. HDCP Receivers may be HDCP Repeaters that serve as downstream HDCP Transmitters emitting the HDCP Content further downstream to one or more additional HDCP Receivers. High-bandwidth Digital Content Protection System Mapping HDCP to HDMI Revision 2.2 13 February, 2013 ("HDMI HDCP 2.2") at 5.

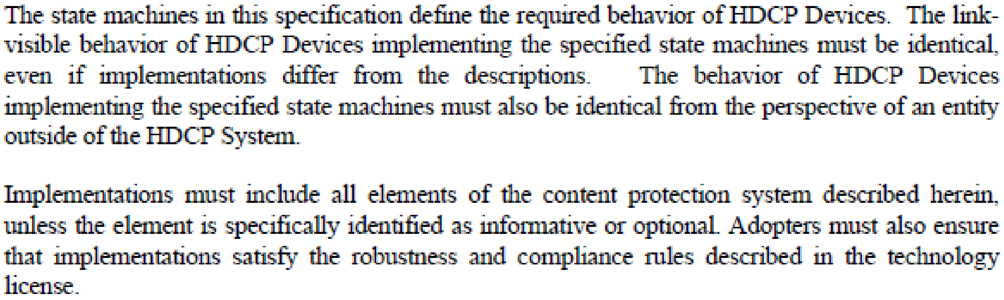


There are three elements of the content protection system. Each element plays a specific role in the system. First, there is the authentication protocol, through which the HDCP Transmitter verifies that a given HDCP Receiver is licensed to receive HDCP Content. The authentication protocol is implemented between the HDCP Transmitter and its corresponding downstream HDCP Receiver. With the legitimacy of the HDCP Receiver determined, encrypted HDCP Content is transmitted between the two devices based on shared secrets established during the authentication protocol. This prevents eavesdropping devices from utilizing the content. Finally, in the event that legitimate devices are compromised to permit unauthorized use of HDCP Content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of HDCP Content.

This document contains chapters describing in detail the requirements of each of these elements. In addition, a chapter is devoted to describing the cipher structure that is used in the encryption of HDCP Content.

Id. at 9.

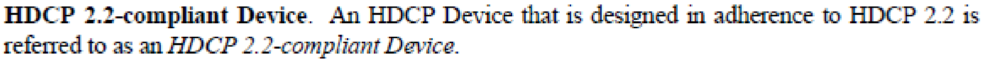
The Accused Product is an HDCP Device, and more specifically an HDCP 2.2-compliant Device, capable of functioning as an HDCP Transmitter and that implements required functionality of HDMI HDCP 2.2 including the functions required by the HDCP Transmitter State Diagram.



The state machines in this specification define the required behavior of HDCP Devices. The link-visible behavior of HDCP Devices implementing the specified state machines must be identical, even if implementations differ from the descriptions. The behavior of HDCP Devices implementing the specified state machines must also be identical from the perspective of an entity outside of the HDCP System.

Implementations must include all elements of the content protection system described herein, unless the element is specifically identified as informative or optional. Adopters must also ensure that implementations satisfy the robustness and compliance rules described in the technology license.

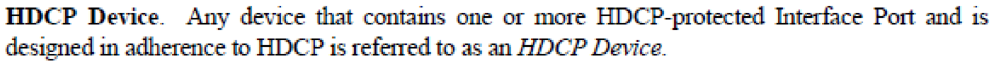
Id. at 5.



HDCP 2.2-compliant Device. An HDCP Device that is designed in adherence to HDCP 2.2 is referred to as an *HDCP 2.2-compliant Device*.

Id. at 6.

"17. A system for controlling the transmission of protected content from a content provider to a requesting device, the content provider comprising:"



HDCP Device. Any device that contains one or more HDCP-protected Interface Port and is designed in adherence to HDCP is referred to as an *HDCP Device*.

Id. at 7.

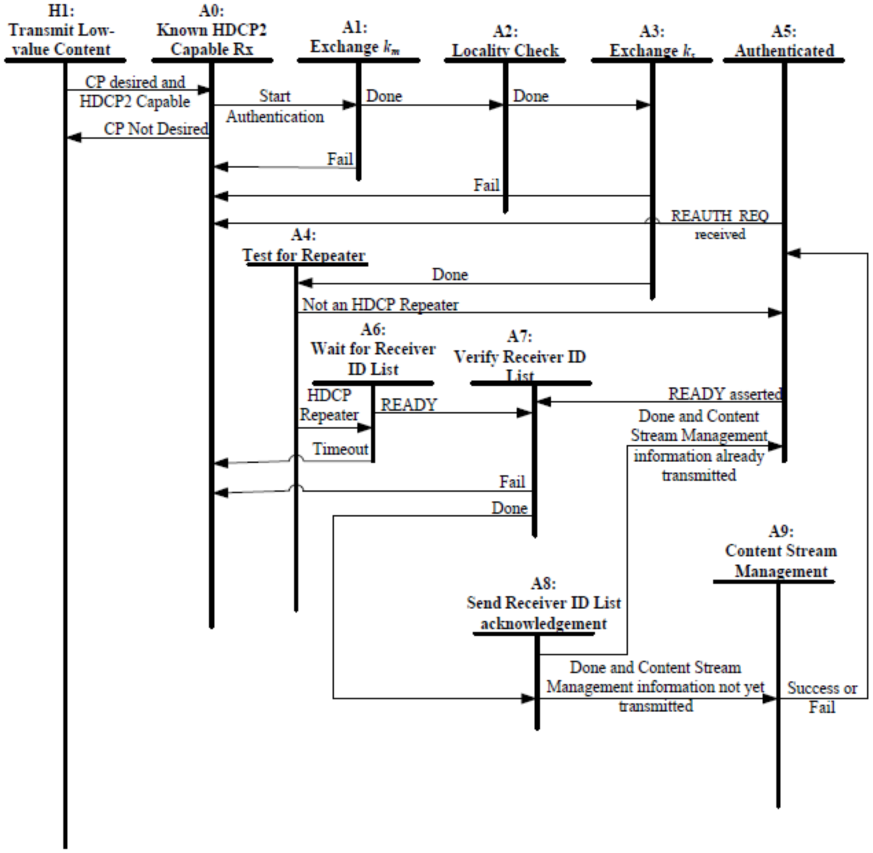
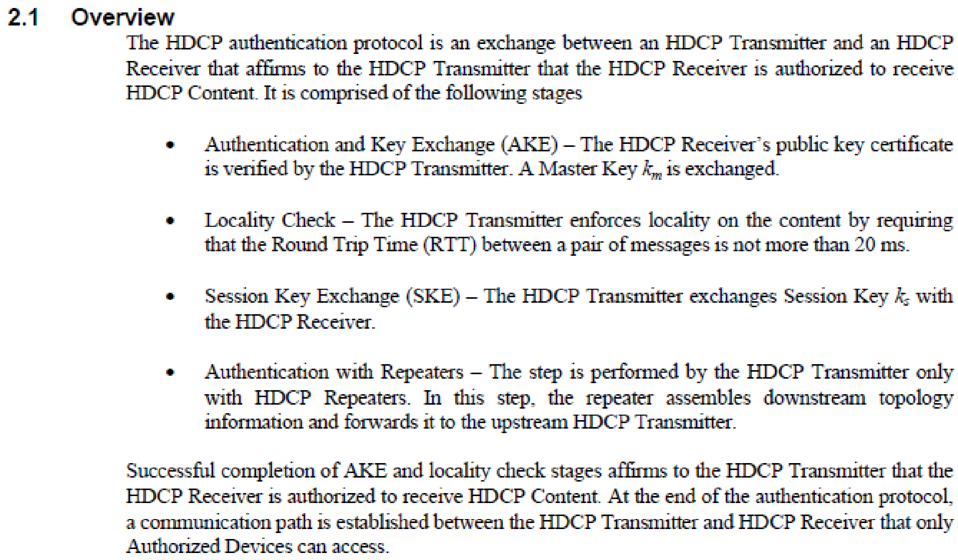


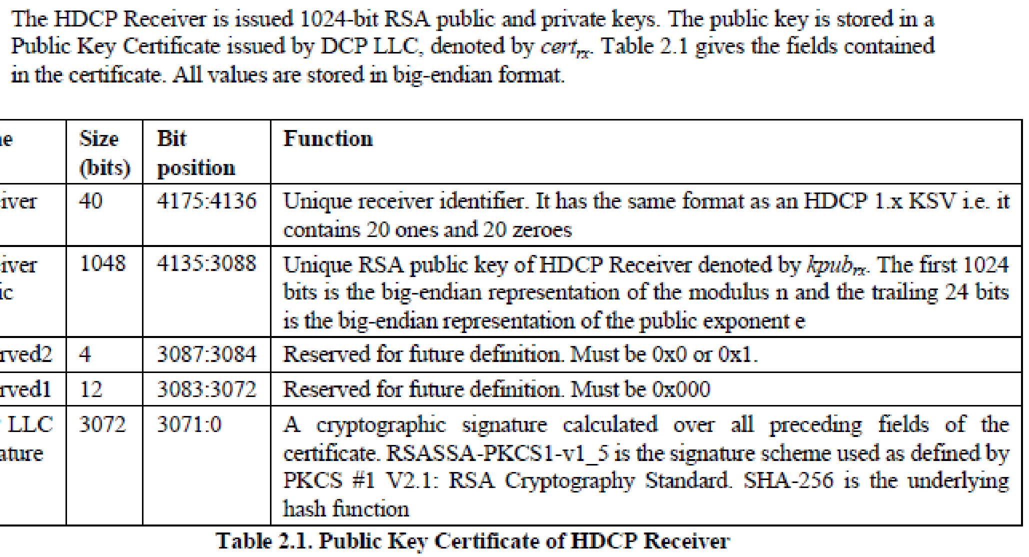
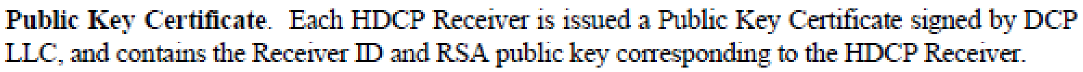
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27-30.

The Accused Product controls delivery of protected content to a second device.

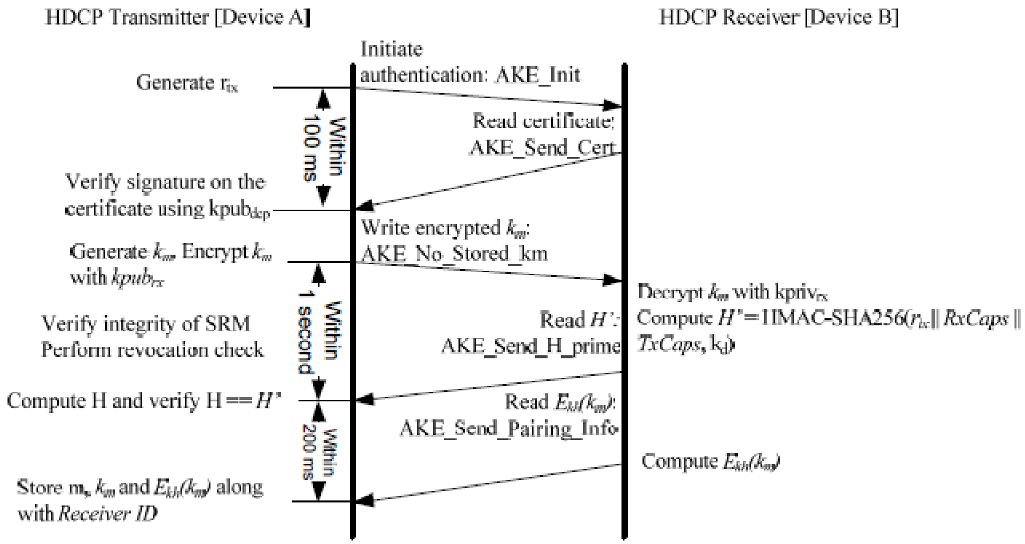
	 **2.1 Overview** The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. *Id*. at 11.

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

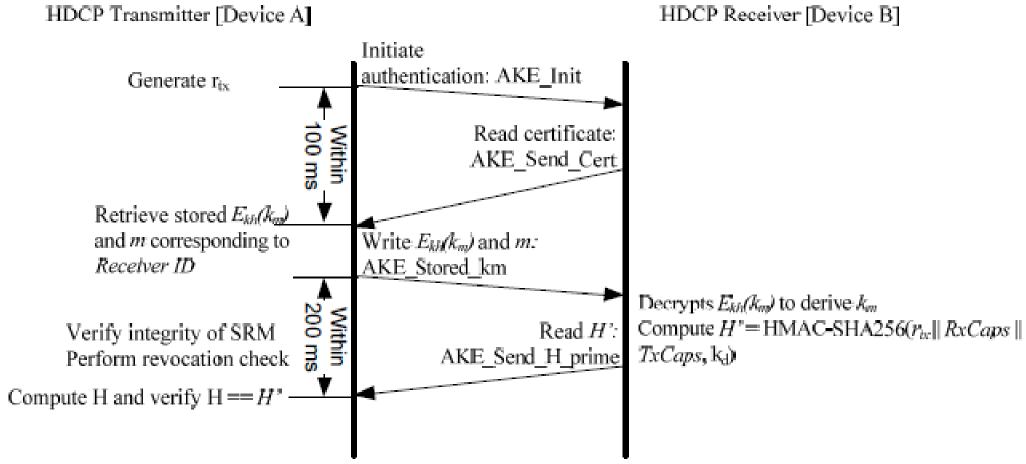
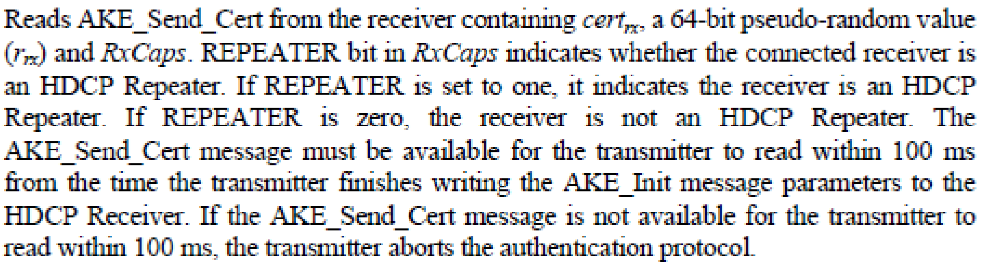
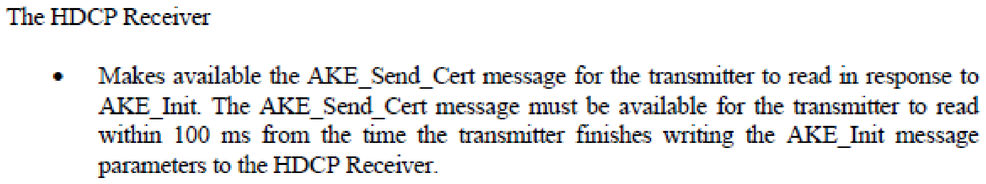
<table>
<tr>
<td>means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;</td>
<td>
<p>The Accused Product comprises means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules.</p>
<p>For example, the Accused Product comprises a receiver and a microprocessor programmed with software for receiving a certificate of the requesting device, <i>e.g.</i>, $cert_{rx}$, as part of the Authentication and Key Exchange (AKE) stage of the HDCP 2.2 protocol, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules.</p>
<p>The certificate, $cert_{rx}$, includes a Receiver ID for the second device, Receiver Public Key for the second device, and a cryptographic signature, amongst other information.</p>

<p>The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.</p>
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
<p>Table 2.1. Public Key Certificate of HDCP Receiver</p>
<p>HDMI HDCP 2.2 at 11.</p>

<p>Public Key Certificate. Each HDCP Receiver is issued a Public Key Certificate signed by DCP LLC, and contains the Receiver ID and RSA public key corresponding to the HDCP Receiver.</p>
<p><i>Id.</i> at 8.</p>
<p>The Accused Product receives the certificate from the second device as part of the AKE stage, irrespective of whether the Accused Product has a Master Key k_m stored corresponding to the Receiver ID.</p>
</td>
</tr>
</table>

U.S. Patent No. 9,436,809: Claim 17

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

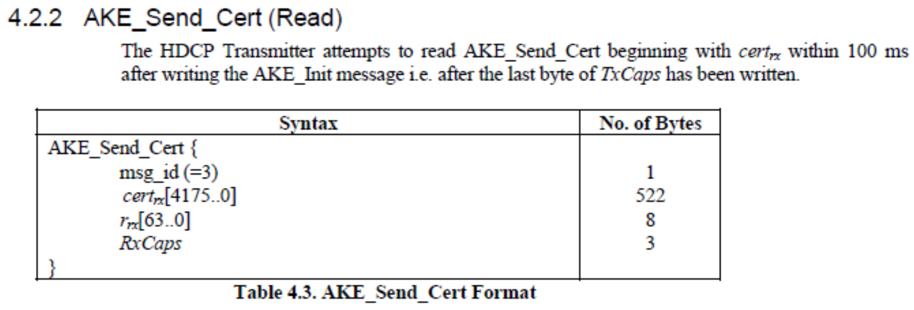
	 **Figure 2.1. Authentication and Key Exchange (Without Stored k_m)** *Id.* at 12.

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

	 **Figure 2.2. Authentication and Key Exchange (With Stored k_m)** *Id.* The Accused Product receives the certificate from the second device as part of the AKE_Send_Cert message.  Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol. *Id.* at 13.  The HDCP Receiver • Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

Id. at 14.



4.2.2 AKE_Send_Cert (Read)

The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

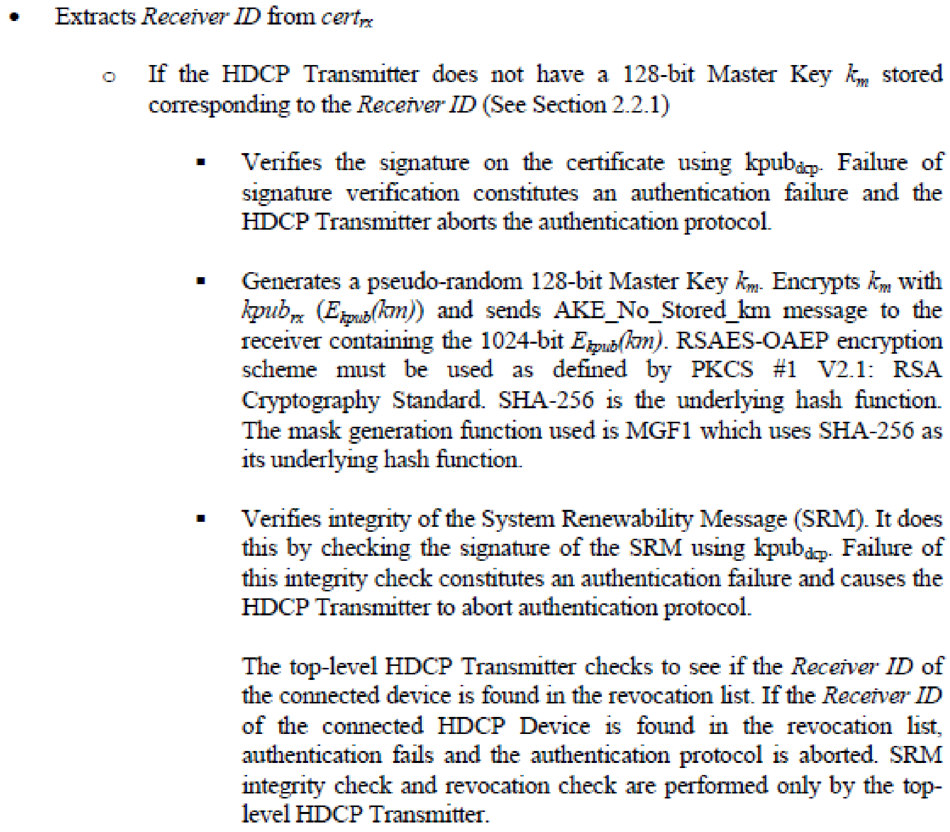
Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

The certificate provides information for use in determining, for example, whether the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"



- Extracts *Receiver ID* from $cert_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using $kpub_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

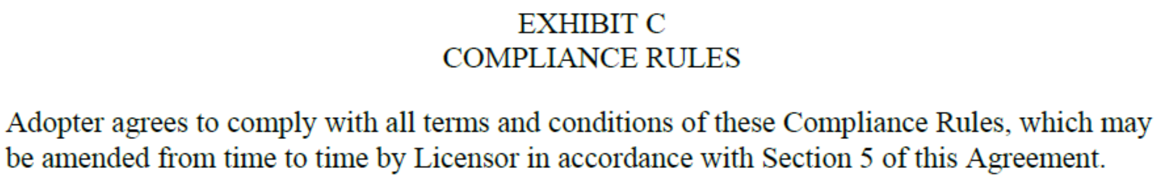
Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2, *available at* https://digital-cp.com/sites/default/files/HDCP%20License%20Agreement_March%206%2C%202017_FOR%20REVIEW%20ONLY.pdf.

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

	 EXHIBIT C COMPLIANCE RULES Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement. *Id*. at Exhibit C. *See also*:

"means for receiving a certificate of the requesting device, the certificate providing information for validating the requesting device as being compliant with a set of compliancy rules;"

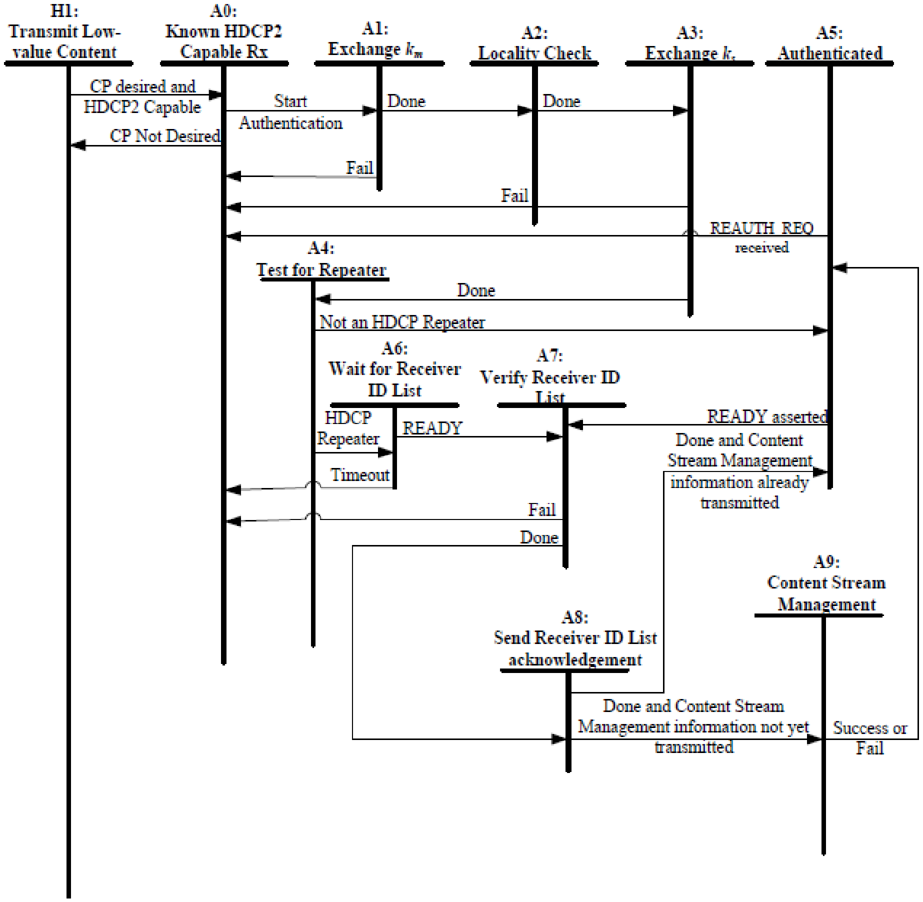
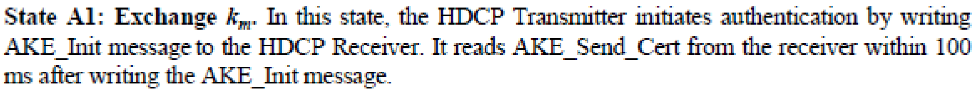


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

HDMI HDCP 2.2 at 27.



State A1: Exchange k_m. In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message.

Id. at 28.

U.S. Patent No. 9,436,809: Claim 17

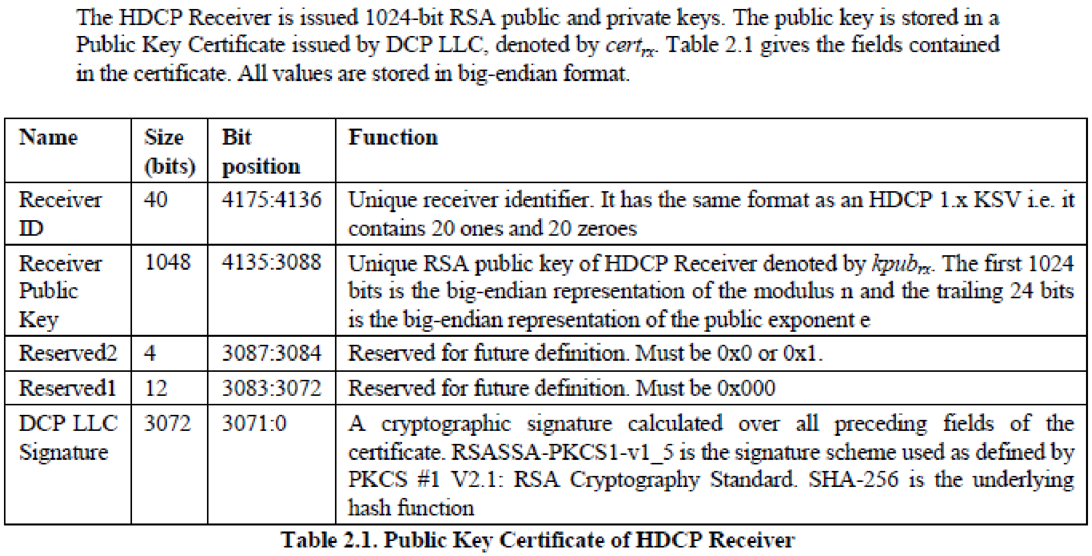
"means for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate;"

<table>
<tr>
<td>means for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate;</td>
<td>

The Accused Product comprises means for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate.

For example, the Accused Product comprises a microprocessor programmed with software for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate.

The Accused Product determines, as part of the Authentication and Key Exchange (AKE) stage, whether the second device is compliant with a set of compliancy rules using the information contained in the certificate, *e.g.*, $cert_{rx}$. For example, $cert_{rx}$ includes a Receiver ID, Receiver Public Key, and a cryptographic signature.



The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

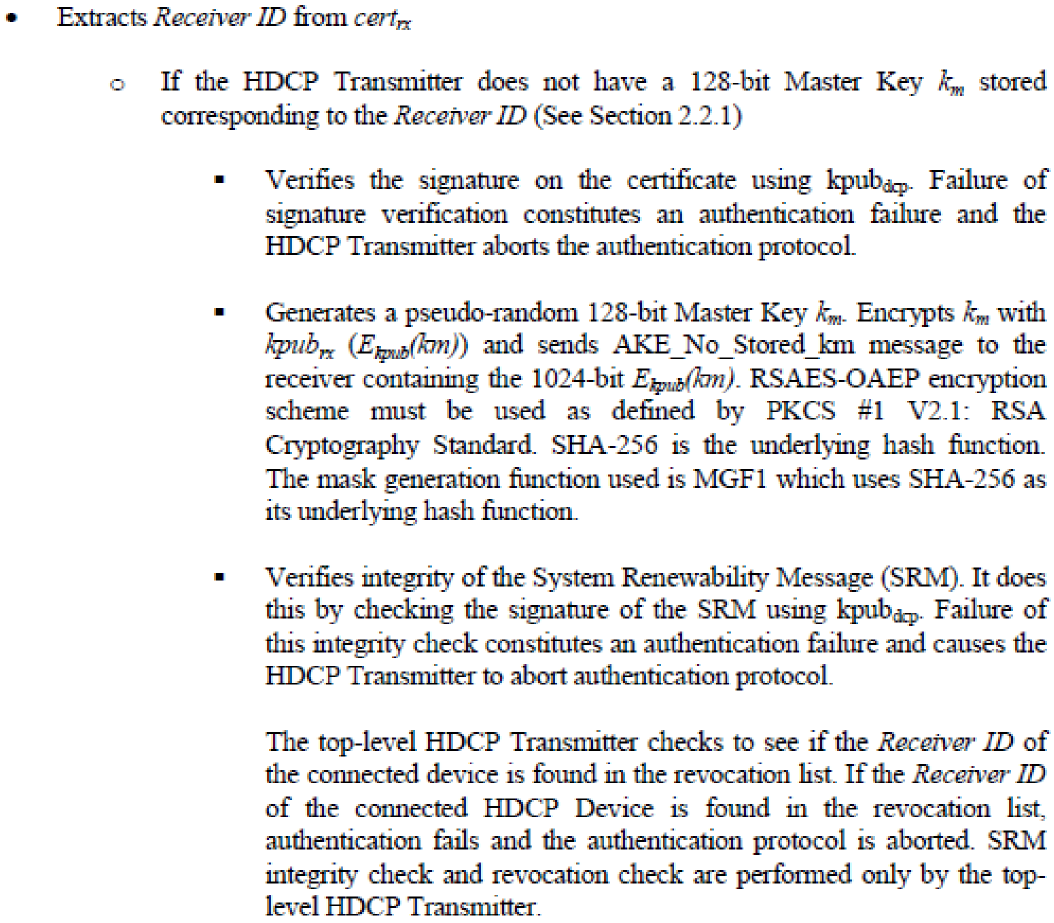
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>

Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.

The Accused Product determines, for example, whether the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.

</td>
</tr>
</table>



- Extracts *Receiver ID* from $cert_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using $kpub_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2.

"means for validating that the requesting device is compliant with the set of compliancy rules using said information contained in said certificate;"

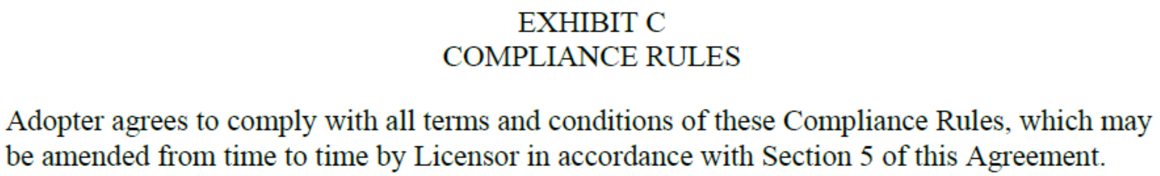


EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

Id. at Exhibit C.

See also:

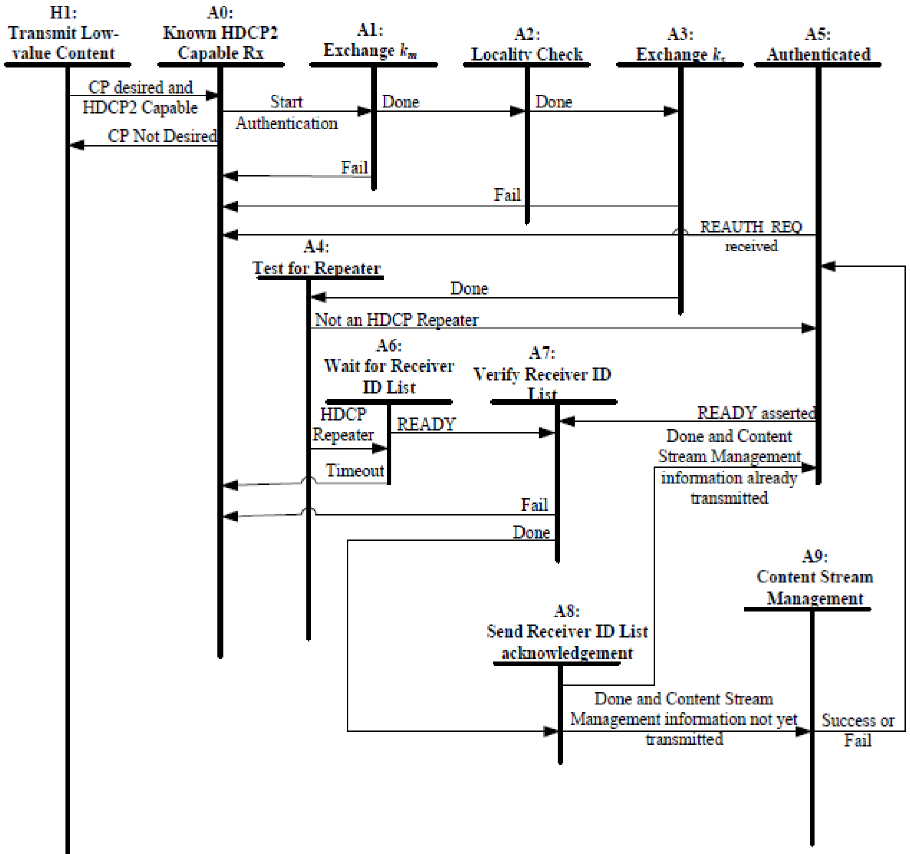
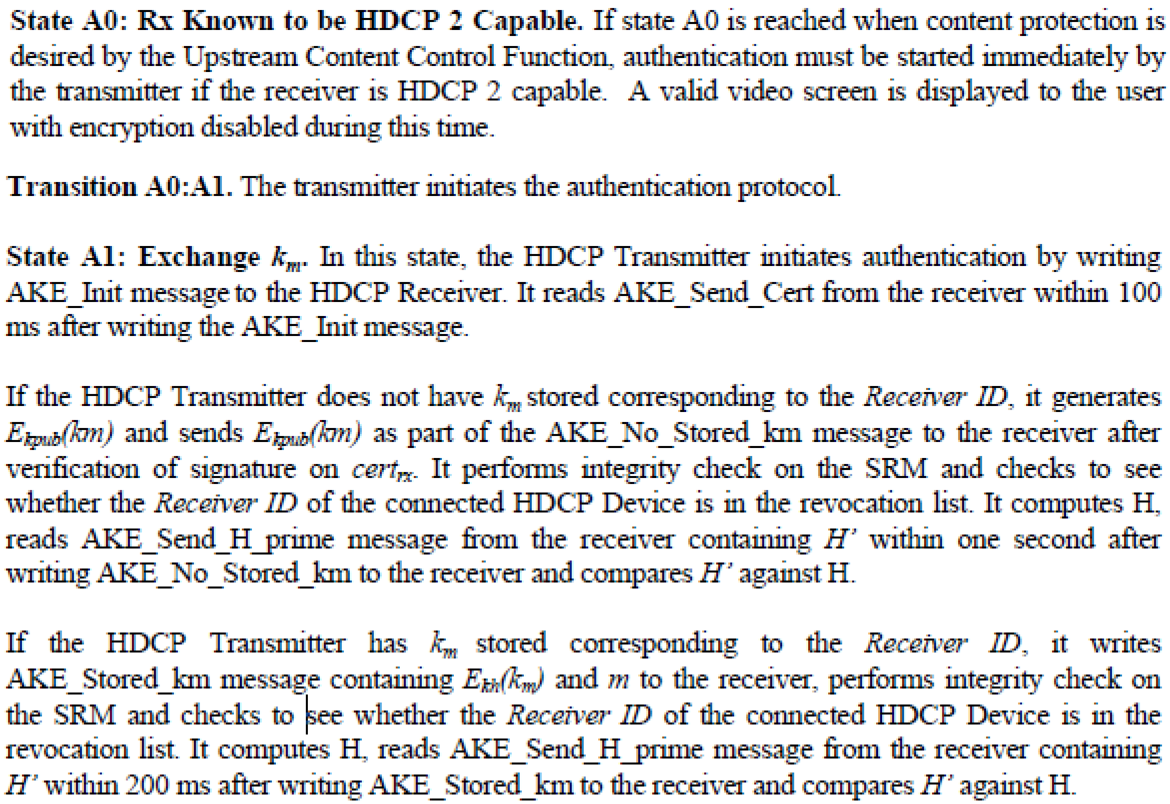


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

HDMI HDCP 2.2 at 27.



State A0: Rx Known to be HDCP 2 Capable. If state A0 is reached when content protection is desired by the Upstream Content Control Function, authentication must be started immediately by the transmitter if the receiver is HDCP 2 capable. A valid video screen is displayed to the user with encryption disabled during this time.

Transition A0:A1. The transmitter initiates the authentication protocol.

State A1: Exchange k_m. In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message.

If the HDCP Transmitter does not have k_m stored corresponding to the *Receiver ID*, it generates $E_{kpub}(km)$ and sends $E_{kpub}(km)$ as part of the AKE_No_Stored_km message to the receiver after verification of signature on $cert_{rx}$. It performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within one second after writing AKE_No_Stored_km to the receiver and compares *H'* against H.

If the HDCP Transmitter has k_m stored corresponding to the *Receiver ID*, it writes AKE_Stored_km message containing $E_{kh}(k_m)$ and m to the receiver, performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within 200 ms after writing AKE_Stored_km to the receiver and compares *H'* against H.

Id. at 28.

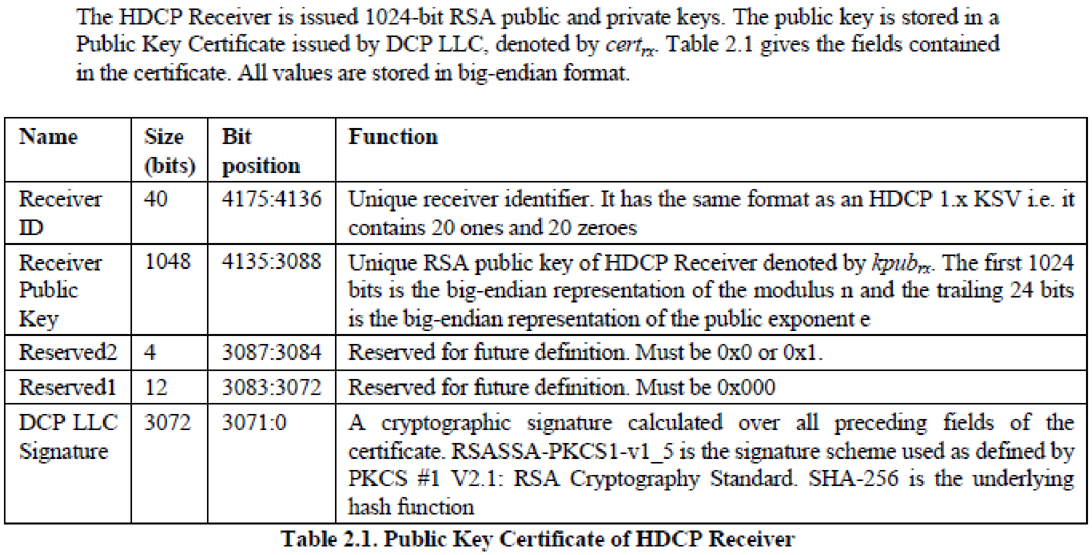
"means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules;"

means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules;

The Accused Product comprises means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules.

For example, the Accused Product comprises a transmitter and a microprocessor programmed with software for transmitting a first signal, *e.g.*, the LC_Init message including r_n, to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules.

The Accused Product provides the LC_Init message including r_n to the second device when the Accused Product determines in the Authentication and Key Exchange (AKE) stage that the certificate, $cert_{rx}$, indicates that the second device is compliant with the set of compliancy rules. For example, the certificate, $cert_{rx}$, includes a Receiver ID, Receiver Public Key, and a cryptographic signature.

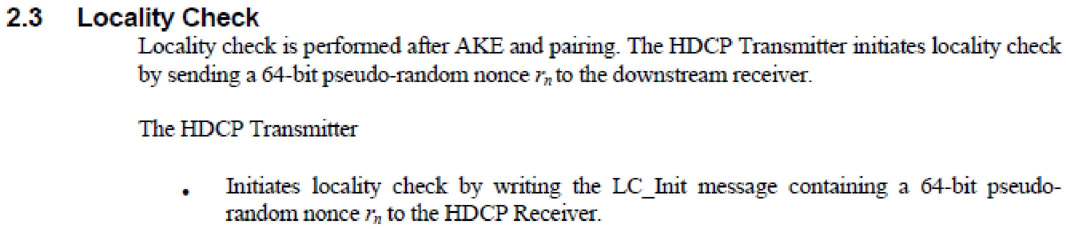


The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

Name	**Size (bits)**	**Bit position**	**Function**
Receiver ID	40	4175:4136	Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes
Receiver Public Key	1048	4135:3088	Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e
Reserved2	4	3087:3084	Reserved for future definition. Must be 0x0 or 0x1.
Reserved1	12	3083:3072	Reserved for future definition. Must be 0x000
DCP LLC Signature	3072	3071:0	A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function

Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.



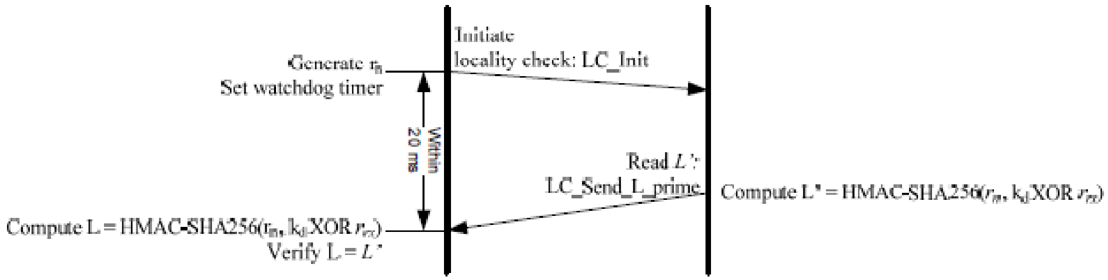
2.3 Locality Check

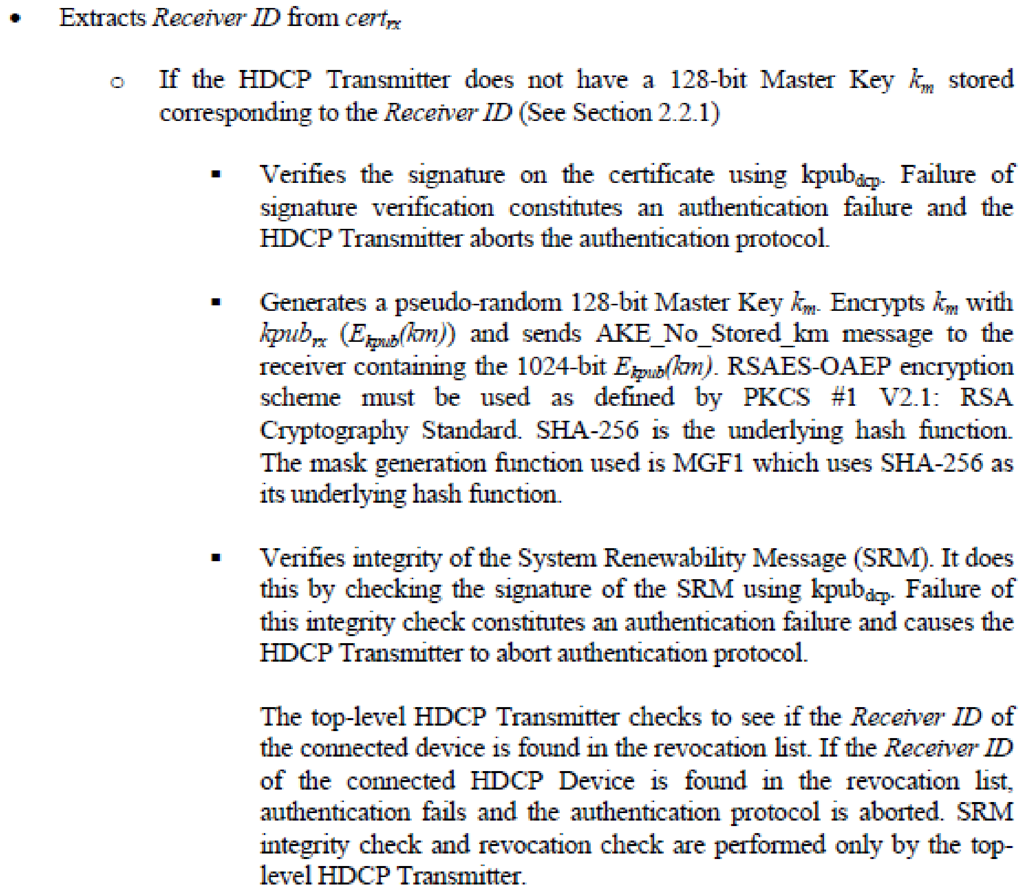
Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.

"means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules;"

	Id. at 16.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** *Id*. at 17. The Accused Product provides the LC_Init message to the second device when, for example, the Accused Product determines that the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.



- Extracts *Receiver ID* from $cert_{rx}$
  - ○ If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - ▪ Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - ▪ Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - ▪ Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using $kpub_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2.

"means for transmitting a first signal to the requesting device at a first time when said requesting device is validated as being compliant with the set of compliancy rules;"

EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

Id. at Exhibit C.

See also:

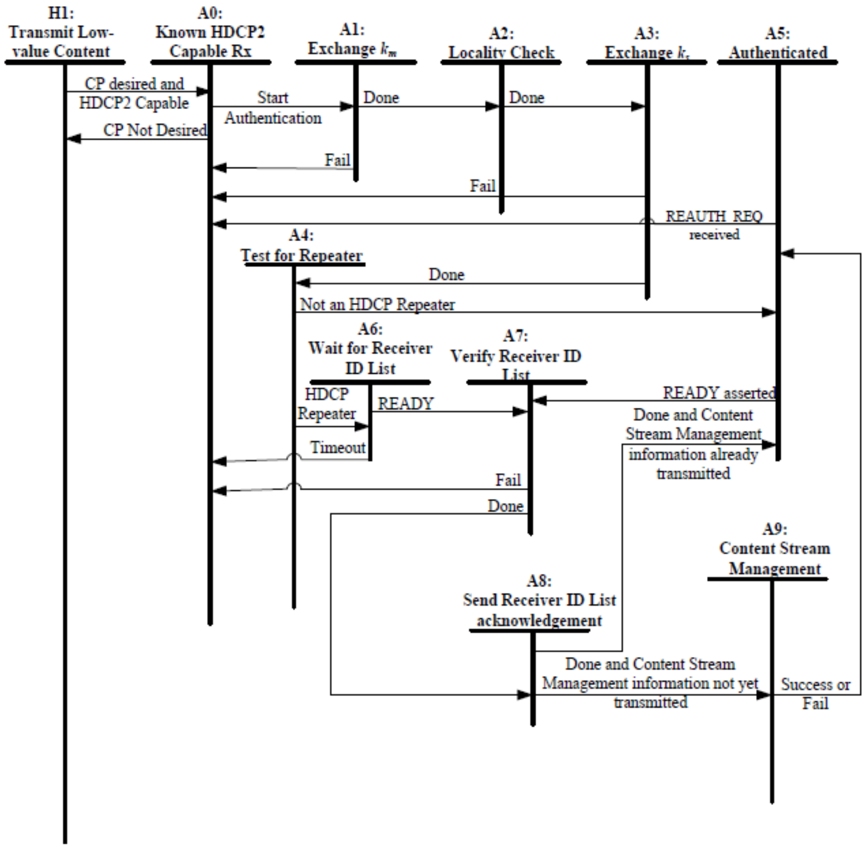
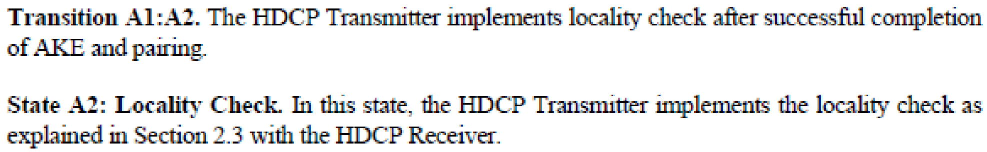
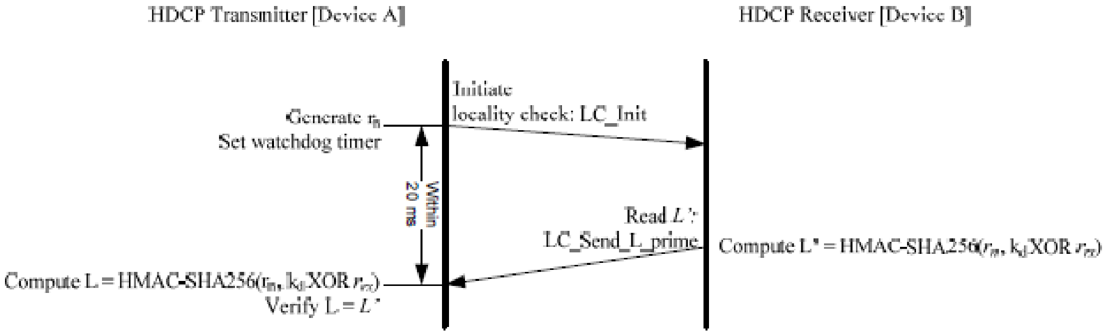
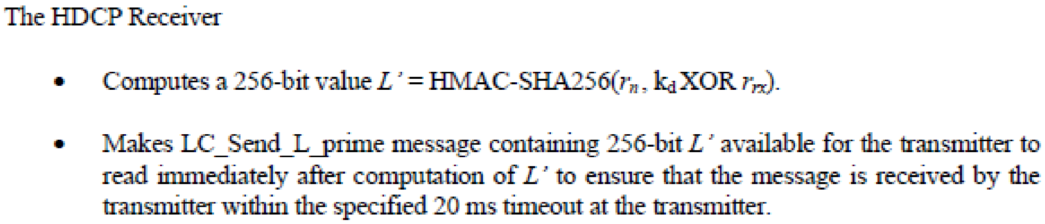
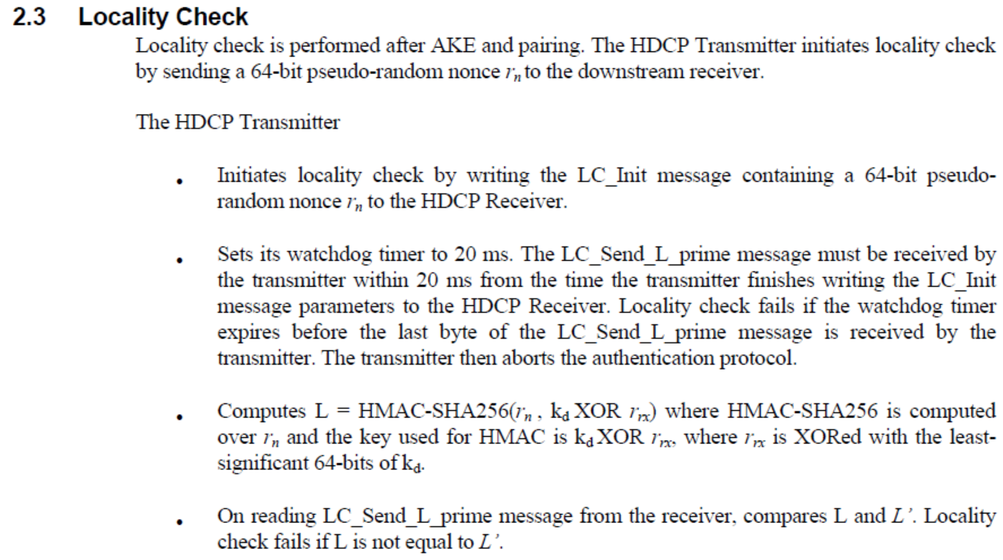


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

	HDMI HDCP 2.2 at 27.  **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id*. at 28.

"means for receiving a second signal at a second time from the requesting device;"

means for receiving a second signal at a second time from the requesting device;	The Accused Product comprises means for receiving a second signal at a second time from the requesting device. For example, the Accused Product comprises a receiver and a microprocessor programmed with software for receiving a second signal, *e.g.*, the LC_Send_L_prime message including L', at a second time from the requesting device.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** HDMI HDCP 2.2 at 17.  The HDCP Receiver - Computes a 256-bit value L' = HMAC-SHA256(r_n, k_d XOR r_{rx}). - Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. *Id.*



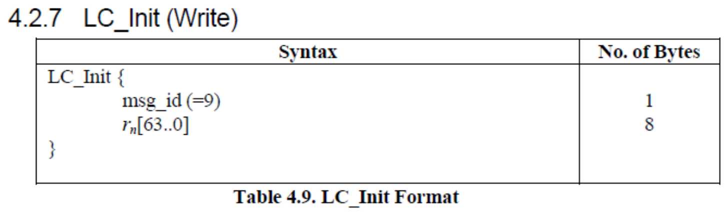
2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

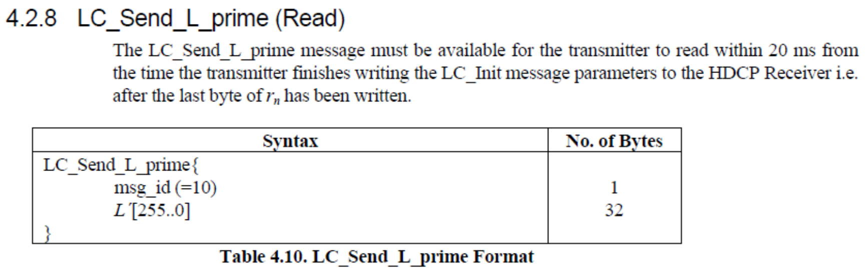


4.2.7 LC_Init (Write)

Syntax	No. of Bytes
LC_Init {	
msg_id (=9)	1
r_n[63..0]	8
}	

Table 4.9. LC_Init Format

Id. at 59.



4.2.8 LC_Send_L_prime (Read)

The LC_Send_L_prime message must be available for the transmitter to read within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver i.e. after the last byte of r_n has been written.

Syntax	No. of Bytes
LC_Send_L_prime{	
msg_id (=10)	1
L'[255..0]	32
}	

Table 4.10. LC_Send_L_prime Format

Id.

See also:

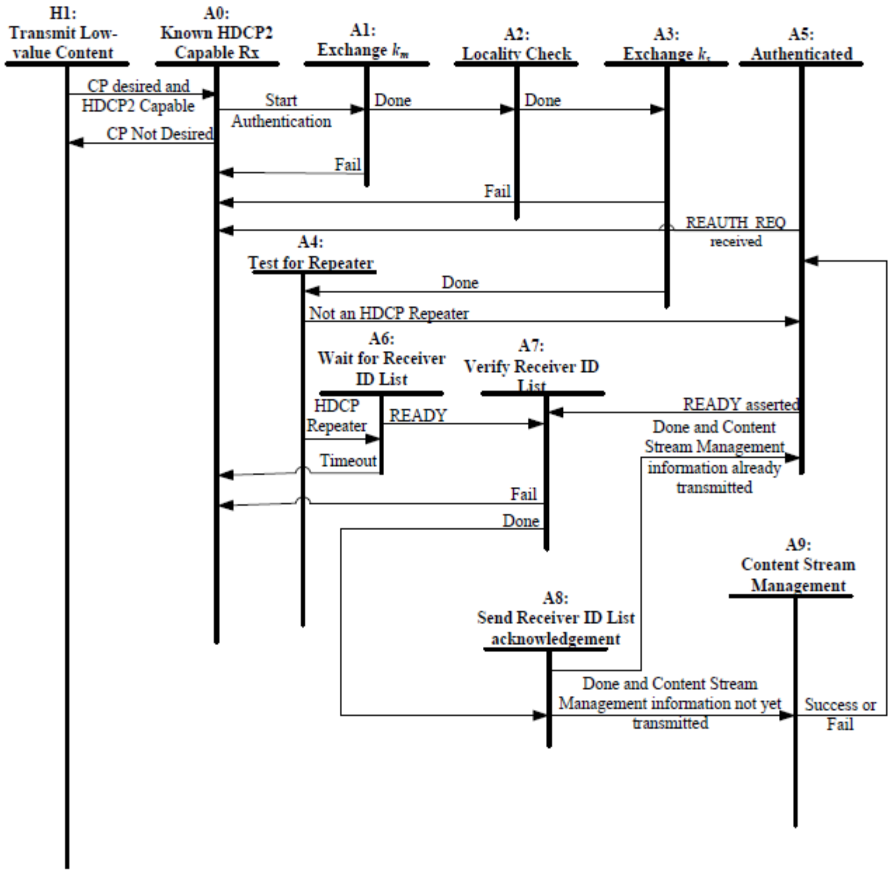


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

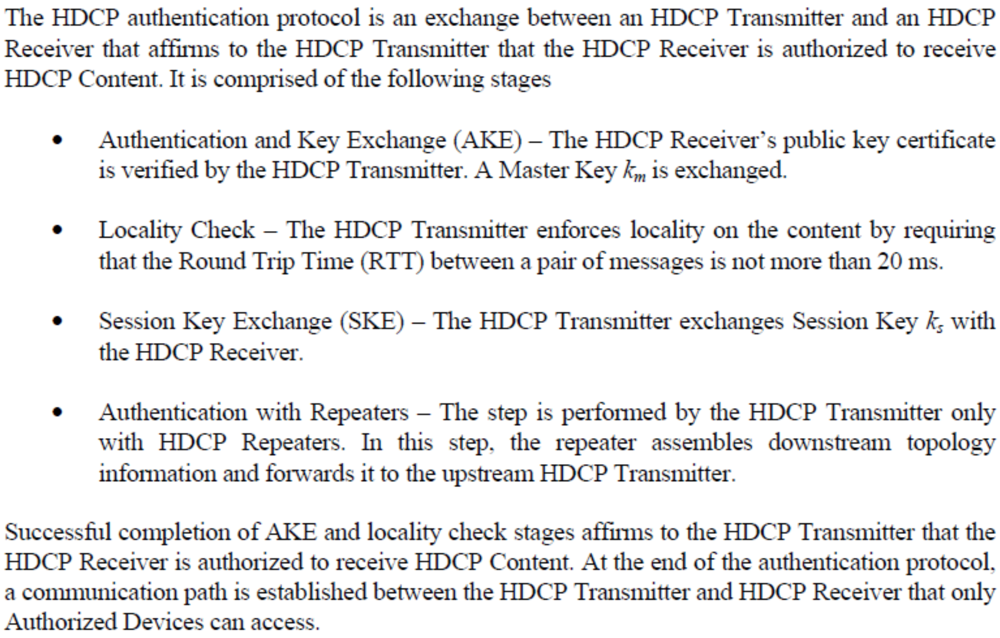
Id. at 27.

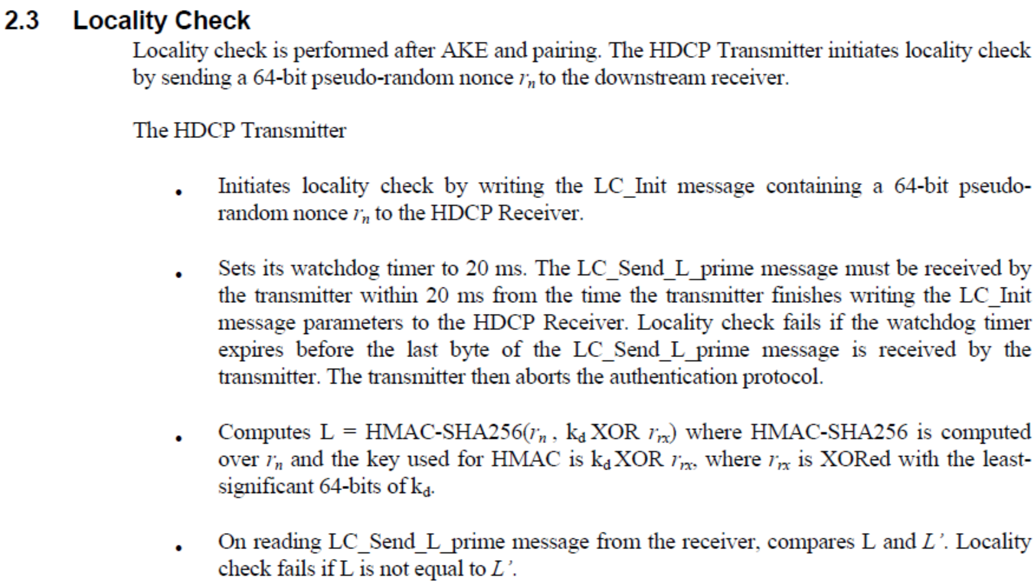
Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."

means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time.	The Accused Product comprises means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the Accused Product, and a time difference between the first time and the second time is less than a predetermined time. For example, the Accused Product comprises a transmitter and a microprocessor programmed with software for providing the protected content to the requesting device after determining the second signal, *e.g.*, L', depends on a secret known to the Accused Product and a time difference between the first time and the second time is less than a predetermined time. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Accused Product provides protected content to the requesting device when, as part of the Locality Check: the L' received via the LC_Send_L_prime message depends on a secret (as determined by matching L' to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$,

	each of which is based on the Master Key, k_m)); and a time between the providing of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than a predetermined time of 20 ms.  **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter - Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. - Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. - Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. - On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id.* at 16.

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."

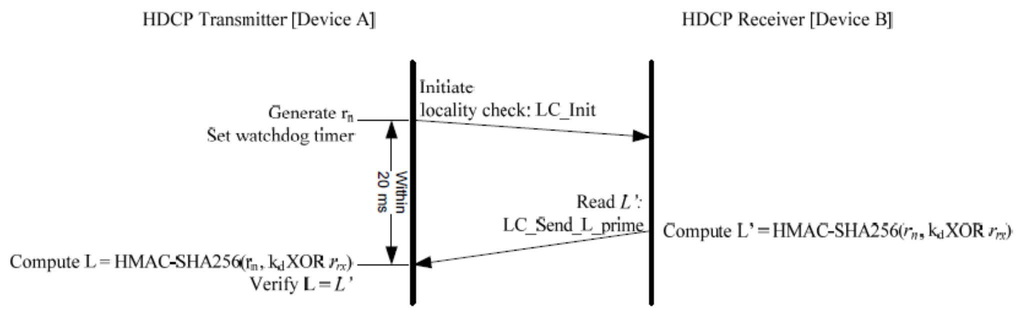


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

The second signal, *e.g.*, L', is derived from a secret.

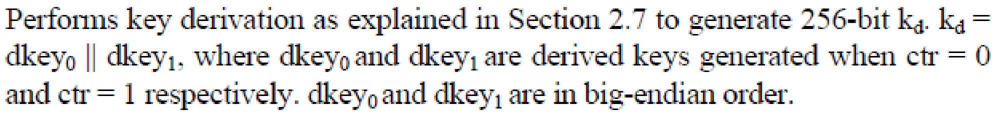
The value of L' is derived from k_d.

Compute $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$

Id.

The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

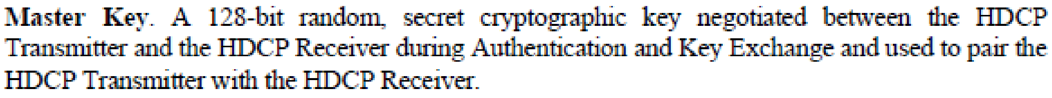
"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."



Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. $k_d = dkey_0 \| dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 14-15.

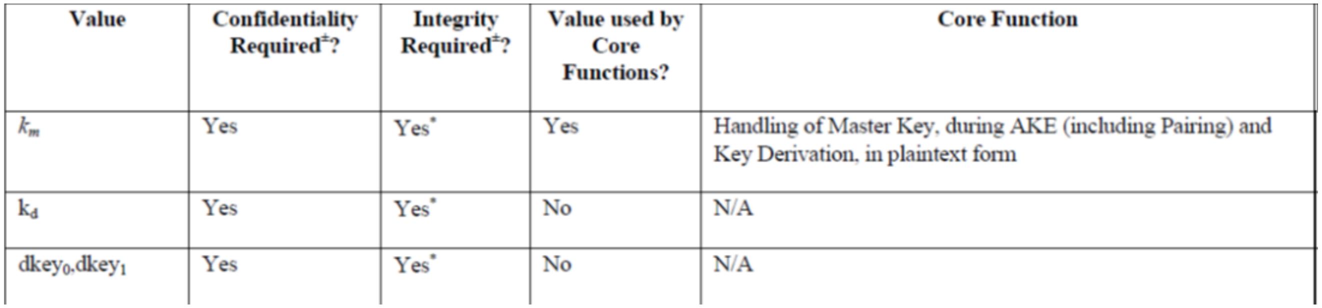
Id. at 25.



Master Key. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver.

Id. at 8.

Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret.

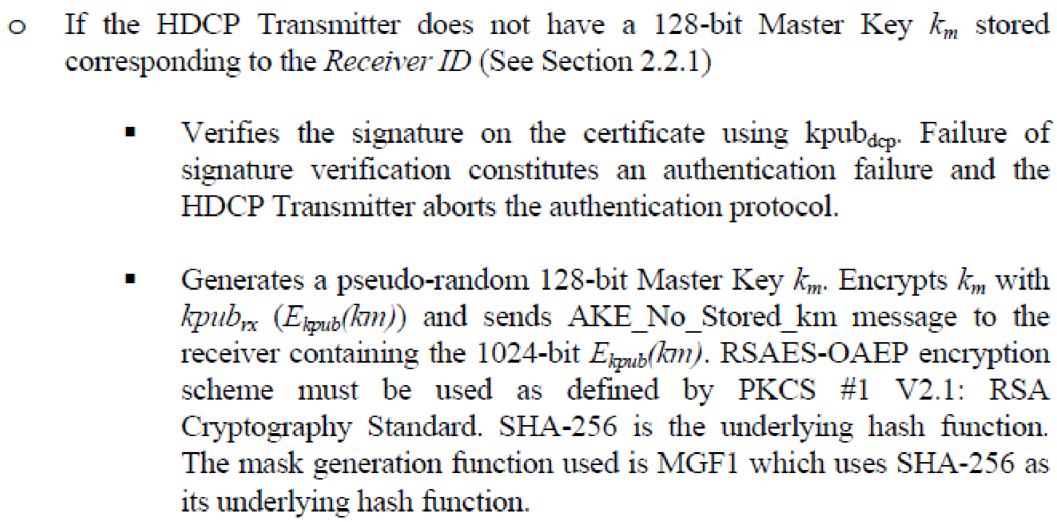
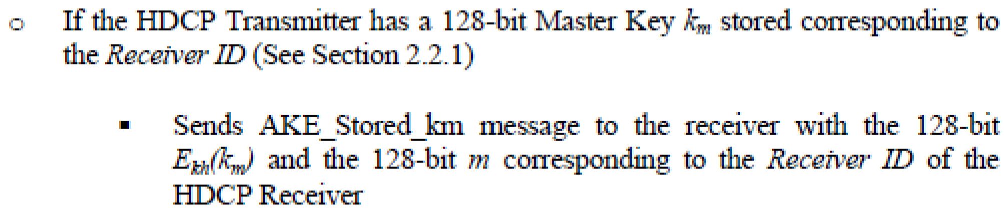


Value	**Confidentiality Required$^{\pm}$?**	**Integrity Required$^{\pm}$?**	**Value used by Core Functions?**	**Core Function**
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
$dkey_0$,$dkey_1$	Yes	Yes*	No	N/A

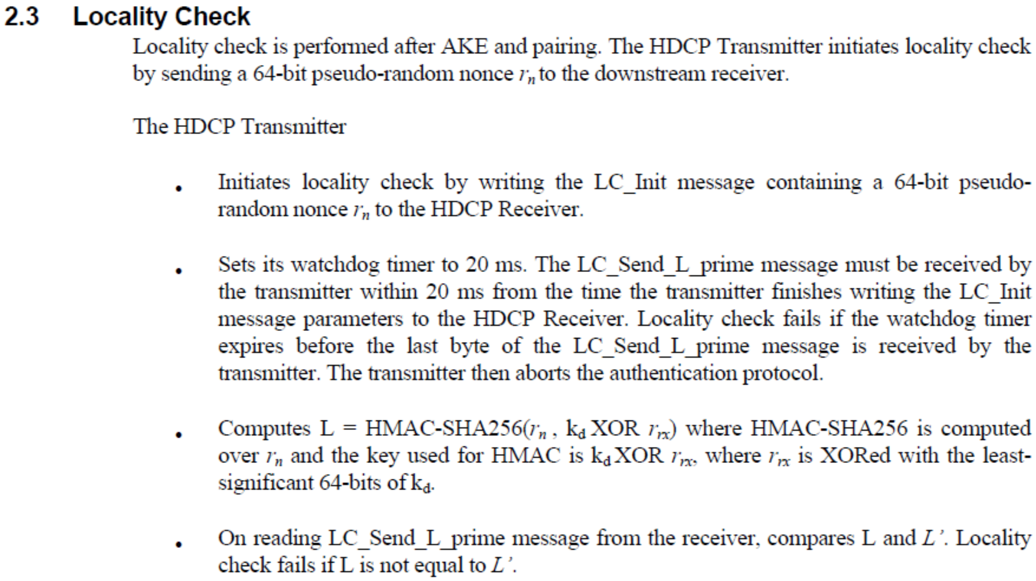
Id. at 67 (abridged).

The Accused Product (transmitter) generates and/or stores the Master Key k_m and thus knows the secret.

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."

	 ○ If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1) ▪ Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol. ▪ Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function. *Id*. at 13.  ○ If the HDCP Transmitter has a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1) ▪ Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit *m* corresponding to the *Receiver ID* of the HDCP Receiver *Id*. at 14. The Accused Product also knows k_d.

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."



2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

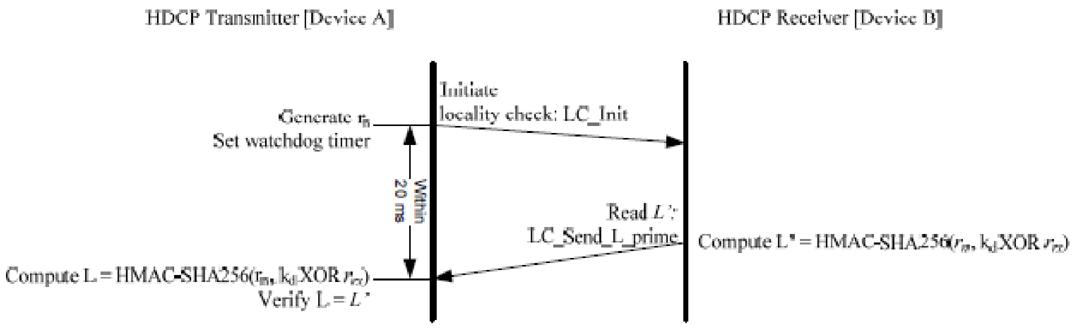
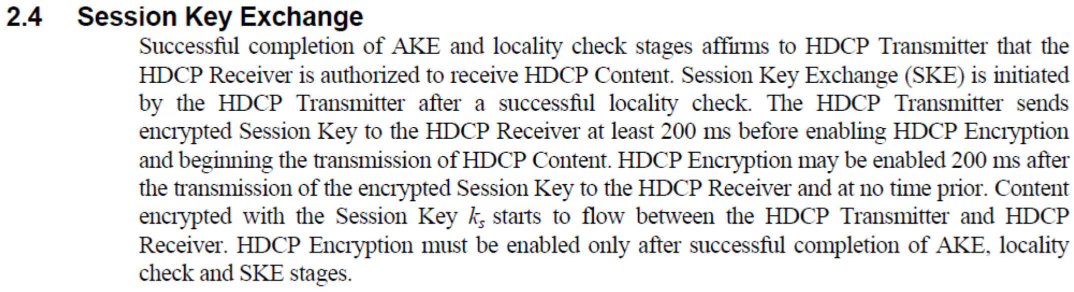


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

Id. at 17.

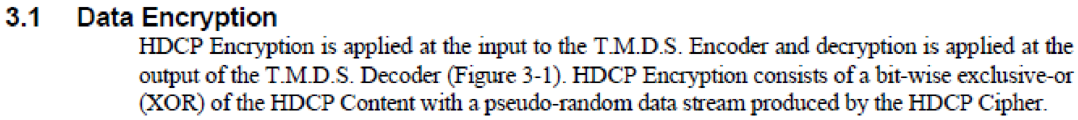
The Accused Product proceeds to session key exchange and providing of the protected content to the second device after successful completion of the AKE stage and Locality Check.



2.4 Session Key Exchange

Successful completion of AKE and locality check stages affirms to HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. Session Key Exchange (SKE) is initiated by the HDCP Transmitter after a successful locality check. The HDCP Transmitter sends encrypted Session Key to the HDCP Receiver at least 200 ms before enabling HDCP Encryption and beginning the transmission of HDCP Content. HDCP Encryption may be enabled 200 ms after the transmission of the encrypted Session Key to the HDCP Receiver and at no time prior. Content encrypted with the Session Key k_s starts to flow between the HDCP Transmitter and HDCP Receiver. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages.

Id. at 17.



3.1 Data Encryption

HDCP Encryption is applied at the input to the T.M.D.S. Encoder and decryption is applied at the output of the T.M.D.S. Decoder (Figure 3-1). HDCP Encryption consists of a bit-wise exclusive-or (XOR) of the HDCP Content with a pseudo-random data stream produced by the HDCP Cipher.

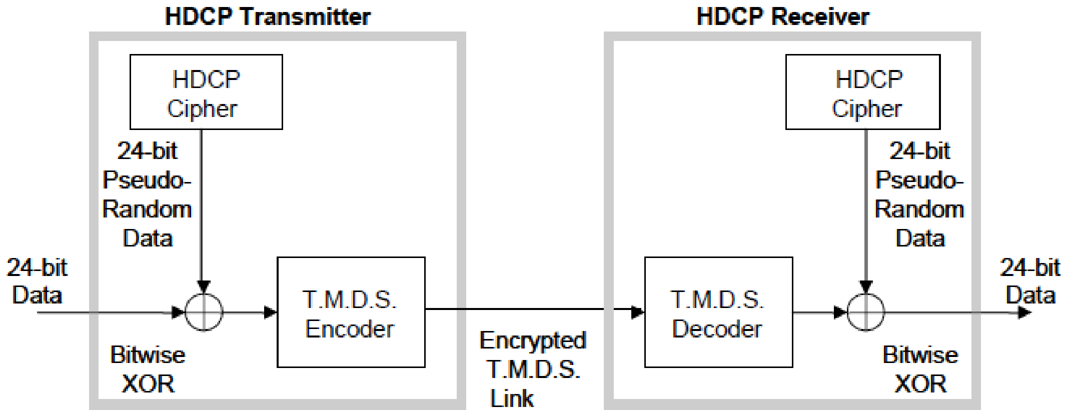


Figure 3-1. HDCP Encryption and Decryption

Id. at 50.

See also:

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."

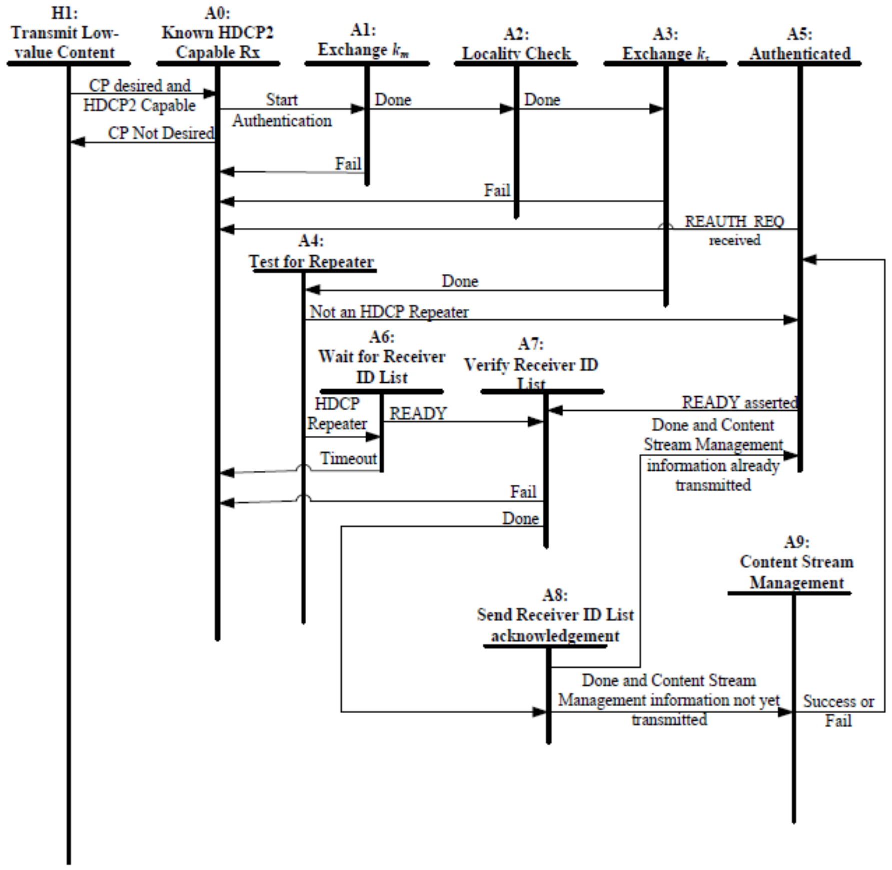
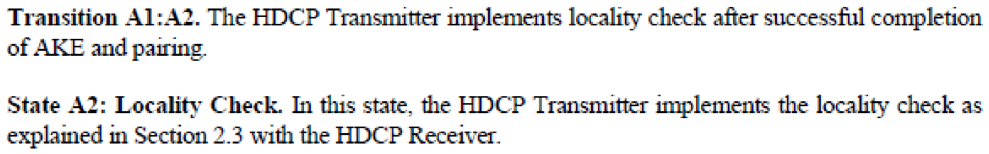


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

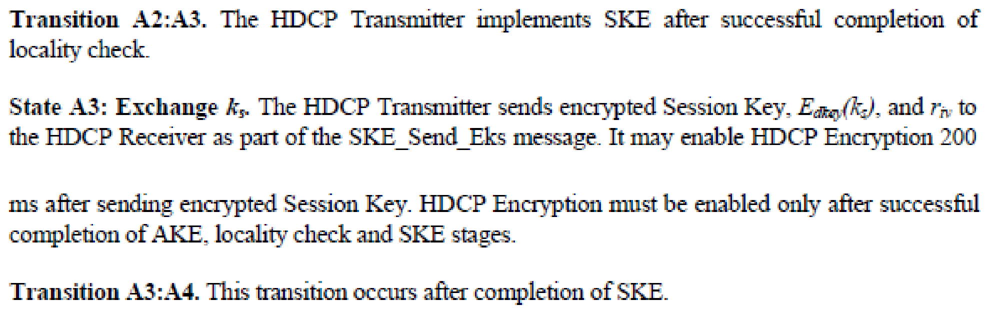


Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"means for providing the protected content to the requesting device after determining the second signal depends on a secret known to the content provider, and a time difference between the first time and the second time is less than a predetermined time."

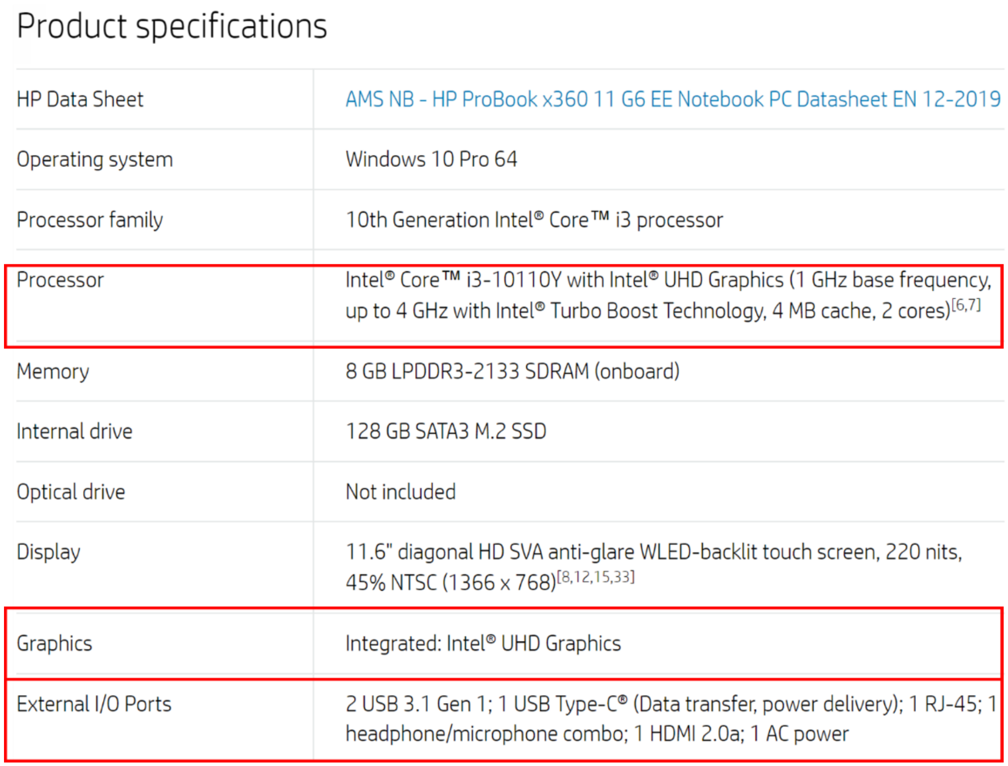
	 **Transition A2:A3.** The HDCP Transmitter implements SKE after successful completion of locality check. **State A3: Exchange k_s.** The HDCP Transmitter sends encrypted Session Key, $E_{dkey}(k_s)$, and r_{rx} to the HDCP Receiver as part of the SKE_Send_Eks message. It may enable HDCP Encryption 200 ms after sending encrypted Session Key. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages. **Transition A3:A4.** This transition occurs after completion of SKE. *Id*. at 28-29.

U.S. Patent No. 9,436,809: Claim 49

"49. A first device for controlling delivery of protected content to a second device, the first device comprising:"

49. A first device for controlling delivery of protected content to a second device, the first device comprising:	Each of the HP Product and the Intel Product is a first device for controlling delivery of protected content to a second device, and is referred to herein as an "Accused Product." For example, the HP Product is an HDMI transmitter with HDCP 2.2 for controlling delivery of protected content to another device, such as an HDMI receiver with HDCP 2.2.  HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. The HP Product includes an HDMI 2.0a port and a 10th Generation Intel® Core™ i3-10110Y Processor (the "Intel Processor") integrated with the Intel UHD Graphics 615 graphics processor (the "Intel GPU") that enable delivery of protected content to another device.

"49. A first device for controlling delivery of protected content to a second device, the first device comprising:"



Product specifications

HP Data Sheet	AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019
Operating system	Windows 10 Pro 64
Processor family	10th Generation Intel® Core™ i3 processor
Processor	Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]
Memory	8 GB LPDDR3-2133 SDRAM (onboard)
Internal drive	128 GB SATA3 M.2 SSD
Optical drive	Not included
Display	11.6" diagonal HD SVA anti-glare WLED-backlit touch screen, 220 nits, 45% NTSC (1366 x 768)[8,12,15,33]
Graphics	Integrated: Intel® UHD Graphics
External I/O Ports	2 USB 3.1 Gen 1; 1 USB Type-C® (Data transfer, power delivery); 1 RJ-45; 1 headphone/microphone combo; 1 HDMI 2.0a; 1 AC power

Id. *See also* NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.

The Intel Processor supports HDCP 2.2 via HDMI 2.0a.

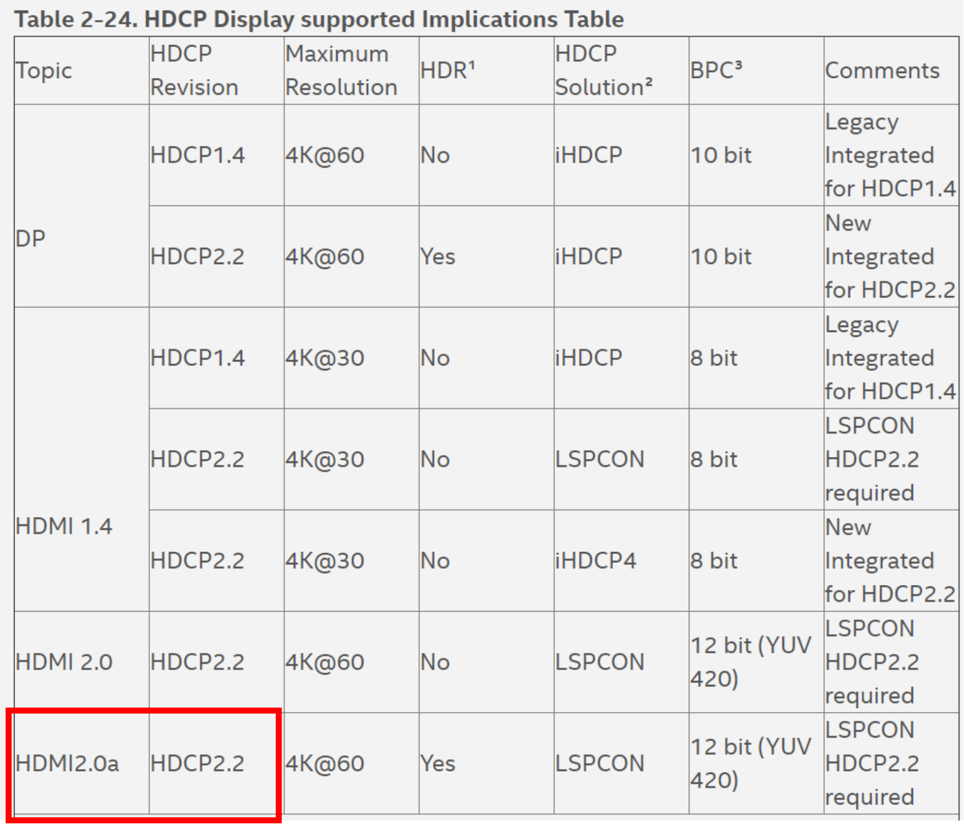
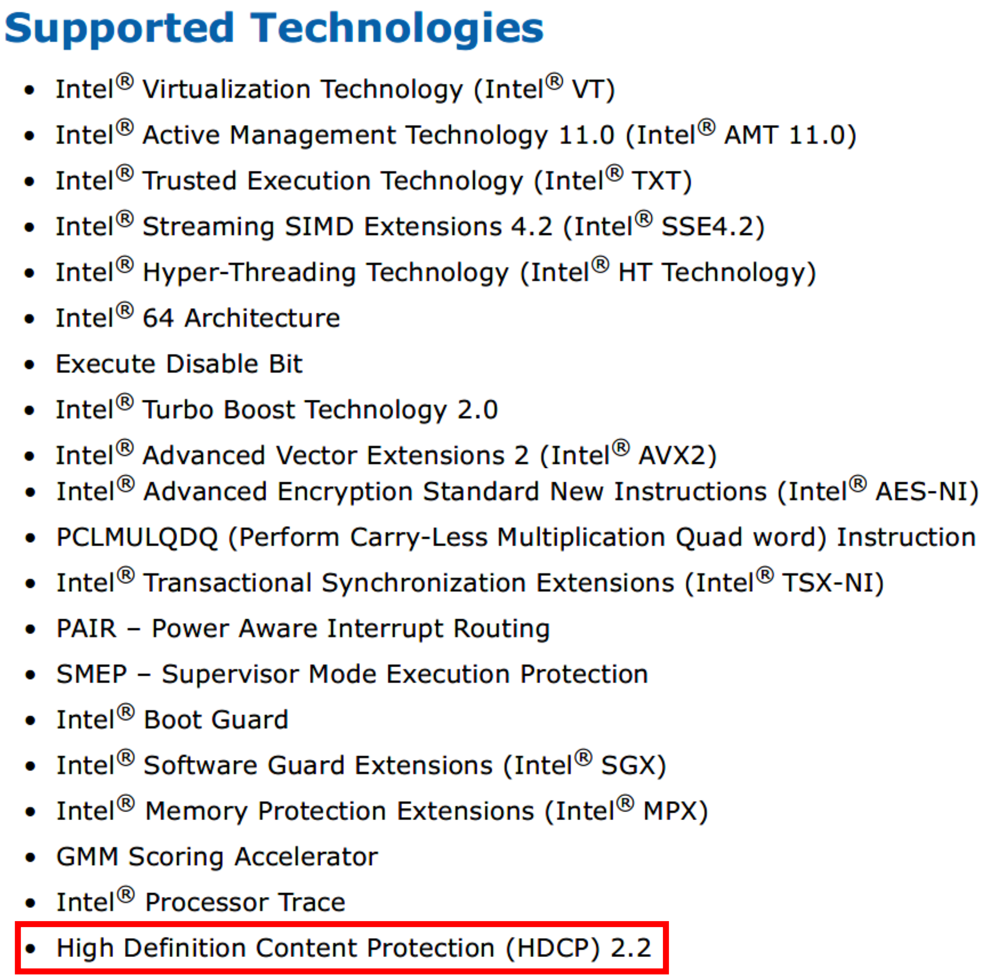
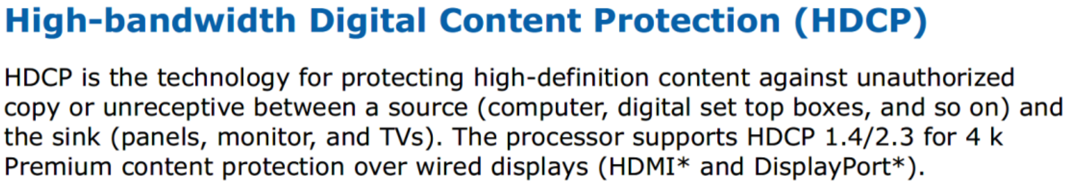


Table 2-24. HDCP Display supported Implications Table

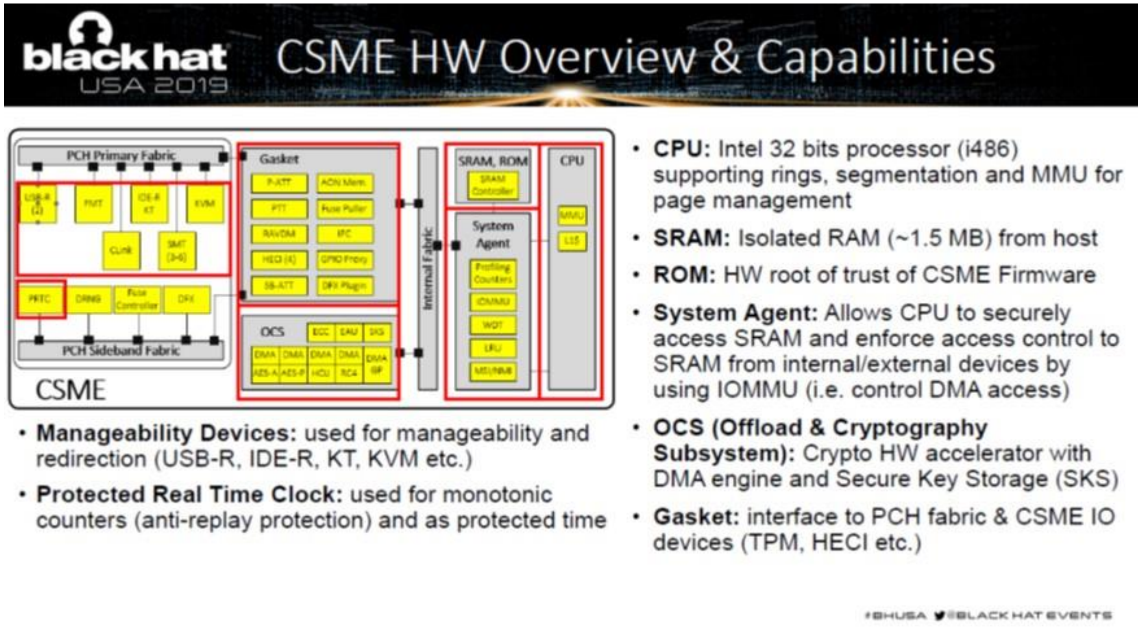
Topic	HDCP Revision	Maximum Resolution	HDR¹	HDCP Solution²	BPC³	Comments
DP	HDCP1.4	4K@60	No	iHDCP	10 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@60	Yes	iHDCP	10 bit	New Integrated for HDCP2.2
HDMI 1.4	HDCP1.4	4K@30	No	iHDCP	8 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@30	No	LSPCON	8 bit	LSPCON HDCP2.2 required
	HDCP2.2	4K@30	No	iHDCP4	8 bit	New Integrated for HDCP2.2
HDMI 2.0	HDCP2.2	4K@60	No	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required
HDMI2.0a	HDCP2.2	4K@60	Yes	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required

Intel, How to enable High Dynamic Range?, https://www.intel.com/content/www/us/en/support/articles/000032112/graphics/graphics-for-7th-generation-intel-processors.html.

	 **Supported Technologies** • Intel® Virtualization Technology (Intel® VT) • Intel® Active Management Technology 11.0 (Intel® AMT 11.0) • Intel® Trusted Execution Technology (Intel® TXT) • Intel® Streaming SIMD Extensions 4.2 (Intel® SSE4.2) • Intel® Hyper-Threading Technology (Intel® HT Technology) • Intel® 64 Architecture • Execute Disable Bit • Intel® Turbo Boost Technology 2.0 • Intel® Advanced Vector Extensions 2 (Intel® AVX2) • Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI) • PCLMULQDQ (Perform Carry-Less Multiplication Quad word) Instruction • Intel® Transactional Synchronization Extensions (Intel® TSX-NI) • PAIR – Power Aware Interrupt Routing • SMEP – Supervisor Mode Execution Protection • Intel® Boot Guard • Intel® Software Guard Extensions (Intel® SGX) • Intel® Memory Protection Extensions (Intel® MPX) • GMM Scoring Accelerator • Intel® Processor Trace • High Definition Content Protection (HDCP) 2.2 Intel, 10th Generation Intel Core Processors, Datasheet, Volume 1 or 2 (Jul. 2020, rev. 5), *available at* https://cdrdv2.intel.com/v1/dl/getContent/615211, at 11-12. “HDCP is the technology for protecting high-definition content against unauthorized copy … between a source … and the sink …. The [Intel] [P]rocessor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*).”

	 **High-bandwidth Digital Content Protection (HDCP)** HDCP is the technology for protecting high-definition content against unauthorized copy or unreceptive between a source (computer, digital set top boxes, and so on) and the sink (panels, monitor, and TVs). The processor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*). *Id.* at 44 Intel's "UHD" processor nomenclature also indicates support for HDCP 2.2: Another change from 7 Gen to 8 Gen will be in the graphics. Intel is upgrading the nomenclature of the integrated graphics from HD 620 to UHD 620, indicating that the silicon is suited for 4K playback and processing. During our pre-briefing it was categorically stated several times that there was no change between the two, however we have since confirmed that the new chips will come with HDCP 2.2 support as standard for DP1.2a, removing the need for an external LSPCON for this feature. Other than this display controller change however, it appears that these new UHD iGPUs are architecturally the same as their HD predecessors. https://www.anandtech.com/show/11738/intel-launches-8th-generation-cpus-starting-with-kaby-lake-refresh-for-15w-mobile. HDCP 2.2 is implemented in Intel-based systems with Core-i series Processors within the Converged Security & Manageability Engine (CSME) also known as the Management Engine (ME). The CSME contains a processor (x86 core) which executes instructions including but not limited to the uKernel/OS, drivers, services, and applications for the CSME.

"49. A first device for controlling delivery of protected content to a second device, the first device comprising:"

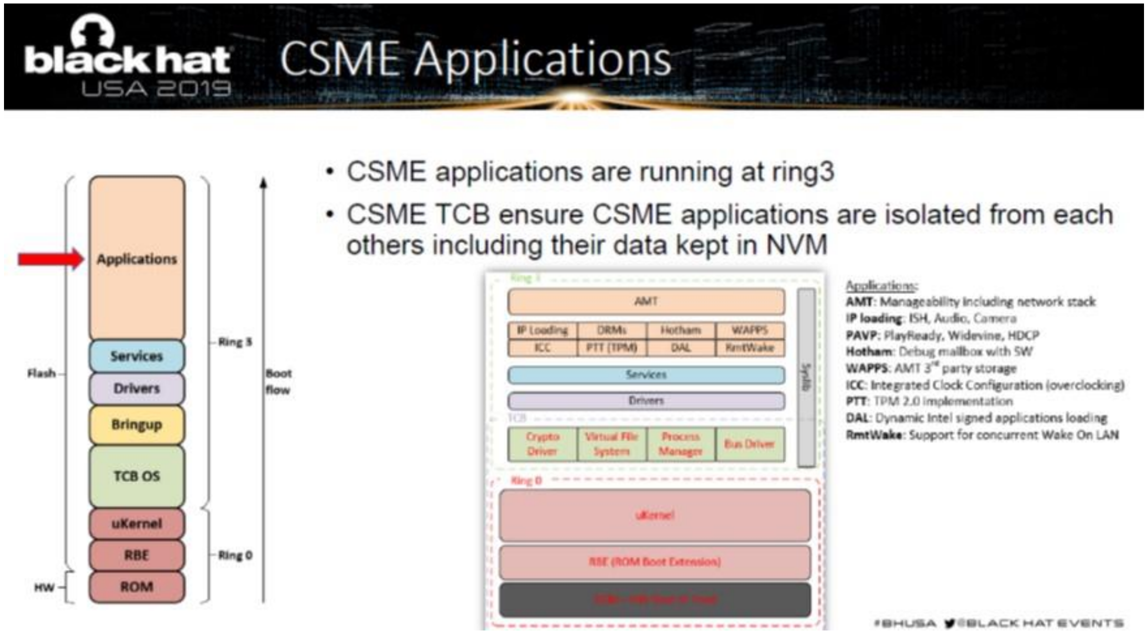


CSME HW Overview & Capabilities

- **Manageability Devices:** used for manageability and redirection (USB-R, IDE-R, KT, KVM etc.)
- **Protected Real Time Clock:** used for monotonic counters (anti-replay protection) and as protected time
- **CPU:** Intel 32 bits processor (i486) supporting rings, segmentation and MMU for page management
- **SRAM:** Isolated RAM (~1.5 MB) from host
- **ROM:** HW root of trust of CSME Firmware
- **System Agent:** Allows CPU to securely access SRAM and enforce access control to SRAM from internal/external devices by using IOMMU (i.e. control DMA access)
- **OCS (Offload & Cryptography Subsystem):** Crypto HW accelerator with DMA engine and Secure Key Storage (SKS)
- **Gasket:** interface to PCH fabric & CSME IO devices (TPM, HECI etc.)

#BHUSA @BLACK HAT EVENTS

Behind the Scenes of Intel Security and Manageability Engine, blackhat USA 2019 ("CSME") at 7.



CSME Applications

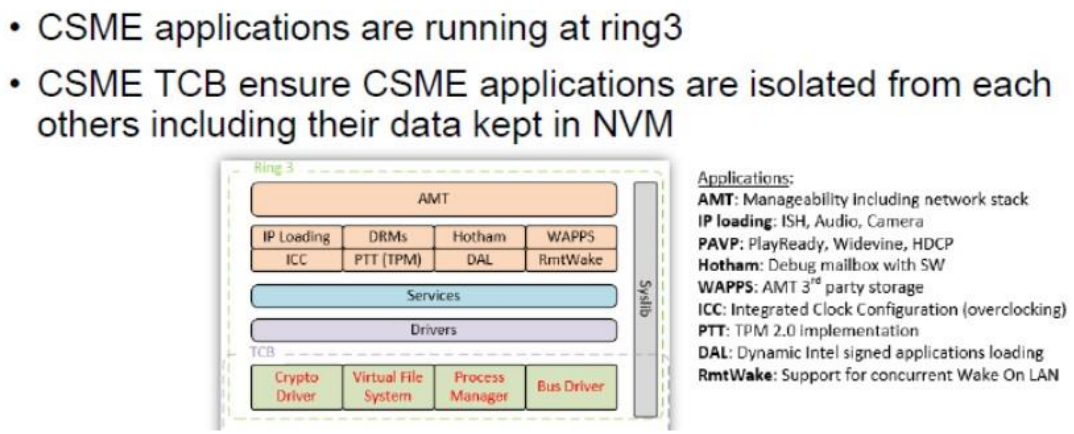
- CSME applications are running at ring3
- CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM

Applications:
AMT: Manageability including network stack
IP loading: ISH, Audio, Camera
PAVP: PlayReady, Widevine, HDCP
Hotham: Debug mailbox with SW
WAPPS: AMT 3rd party storage
ICC: Integrated Clock Configuration (overclocking)
PTT: TPM 2.0 implementation
DAL: Dynamic Intel signed applications loading
RmtWake: Support for concurrent Wake On LAN

#BHUSA @BLACK HAT EVENTS

Id. at 23.

One such application is “PAVP” which provides HDCP capabilities within the Intel processor.

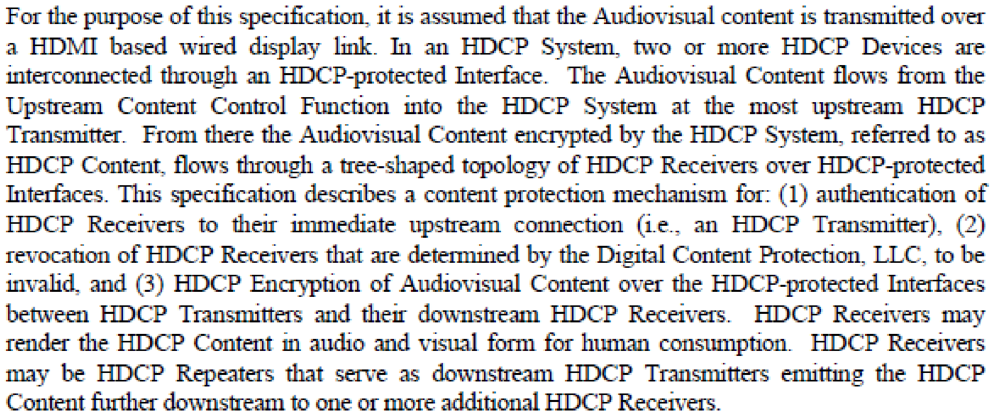


- CSME applications are running at ring3
- CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM

Applications:
AMT: Manageability including network stack
IP loading: ISH, Audio, Camera
PAVP: PlayReady, Widevine, HDCP
Hotham: Debug mailbox with SW
WAPPS: AMT 3rd party storage
ICC: Integrated Clock Configuration (overclocking)
PTT: TPM 2.0 implementation
DAL: Dynamic Intel signed applications loading
RmtWake: Support for concurrent Wake On LAN

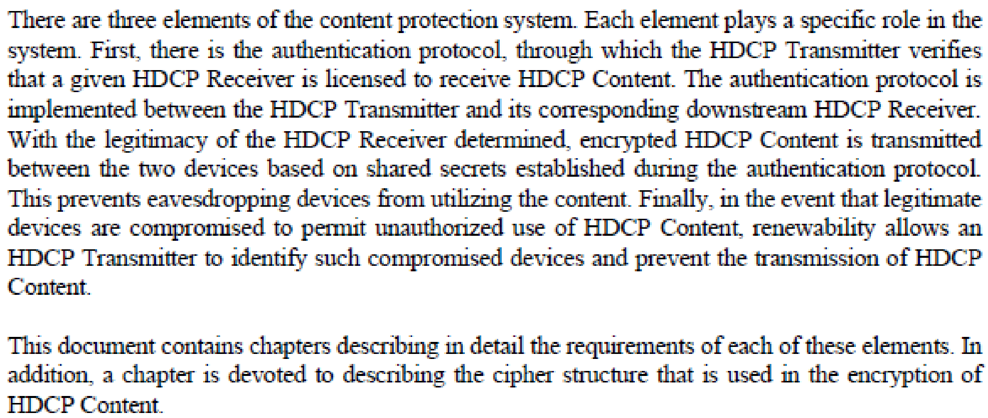
Id.

Upon information and belief, the Accused Product is compliant with the High-bandwidth Digital Content Protection System Revision 2.2 (“HDCP 2.2”) protocol. The Accused Product supports HDCP 2.2 for protecting content between devices.



For the purpose of this specification, it is assumed that the Audiovisual content is transmitted over a HDMI based wired display link. In an HDCP System, two or more HDCP Devices are interconnected through an HDCP-protected Interface. The Audiovisual Content flows from the Upstream Content Control Function into the HDCP System at the most upstream HDCP Transmitter. From there the Audiovisual Content encrypted by the HDCP System, referred to as HDCP Content, flows through a tree-shaped topology of HDCP Receivers over HDCP-protected Interfaces. This specification describes a content protection mechanism for: (1) authentication of HDCP Receivers to their immediate upstream connection (i.e., an HDCP Transmitter), (2) revocation of HDCP Receivers that are determined by the Digital Content Protection, LLC, to be invalid, and (3) HDCP Encryption of Audiovisual Content over the HDCP-protected Interfaces between HDCP Transmitters and their downstream HDCP Receivers. HDCP Receivers may render the HDCP Content in audio and visual form for human consumption. HDCP Receivers may be HDCP Repeaters that serve as downstream HDCP Transmitters emitting the HDCP Content further downstream to one or more additional HDCP Receivers.

High-bandwidth Digital Content Protection System Mapping HDCP to HDMI Revision 2.2 13 February, 2013 (“HDMI HDCP 2.2”) at 5.

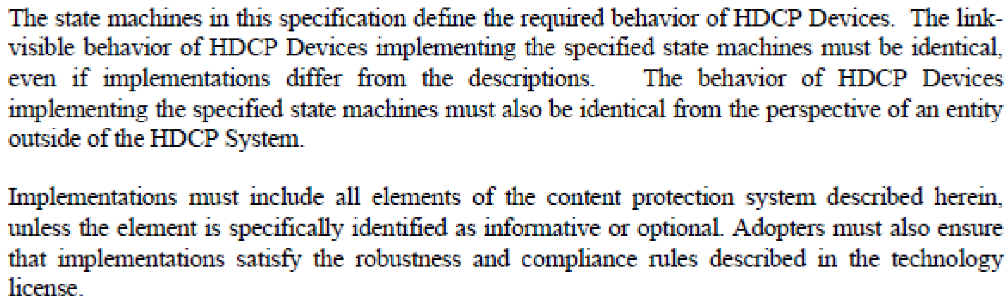


There are three elements of the content protection system. Each element plays a specific role in the system. First, there is the authentication protocol, through which the HDCP Transmitter verifies that a given HDCP Receiver is licensed to receive HDCP Content. The authentication protocol is implemented between the HDCP Transmitter and its corresponding downstream HDCP Receiver. With the legitimacy of the HDCP Receiver determined, encrypted HDCP Content is transmitted between the two devices based on shared secrets established during the authentication protocol. This prevents eavesdropping devices from utilizing the content. Finally, in the event that legitimate devices are compromised to permit unauthorized use of HDCP Content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of HDCP Content.

This document contains chapters describing in detail the requirements of each of these elements. In addition, a chapter is devoted to describing the cipher structure that is used in the encryption of HDCP Content.

Id. at 9.

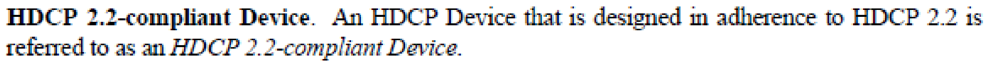
The Accused Product is an HDCP Device, and more specifically an HDCP 2.2-compliant Device, capable of functioning as an HDCP Transmitter and that implements required functionality of HDMI HDCP 2.2 including the functions required by the HDCP Transmitter State Diagram.



The state machines in this specification define the required behavior of HDCP Devices. The link-visible behavior of HDCP Devices implementing the specified state machines must be identical, even if implementations differ from the descriptions. The behavior of HDCP Devices implementing the specified state machines must also be identical from the perspective of an entity outside of the HDCP System.

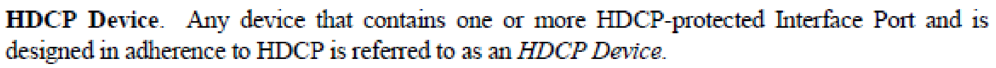
Implementations must include all elements of the content protection system described herein, unless the element is specifically identified as informative or optional. Adopters must also ensure that implementations satisfy the robustness and compliance rules described in the technology license.

Id. at 5.



HDCP 2.2-compliant Device. An HDCP Device that is designed in adherence to HDCP 2.2 is referred to as an *HDCP 2.2-compliant Device*.

Id. at 6.



HDCP Device. Any device that contains one or more HDCP-protected Interface Port and is designed in adherence to HDCP is referred to as an *HDCP Device*.

Id. at 7.

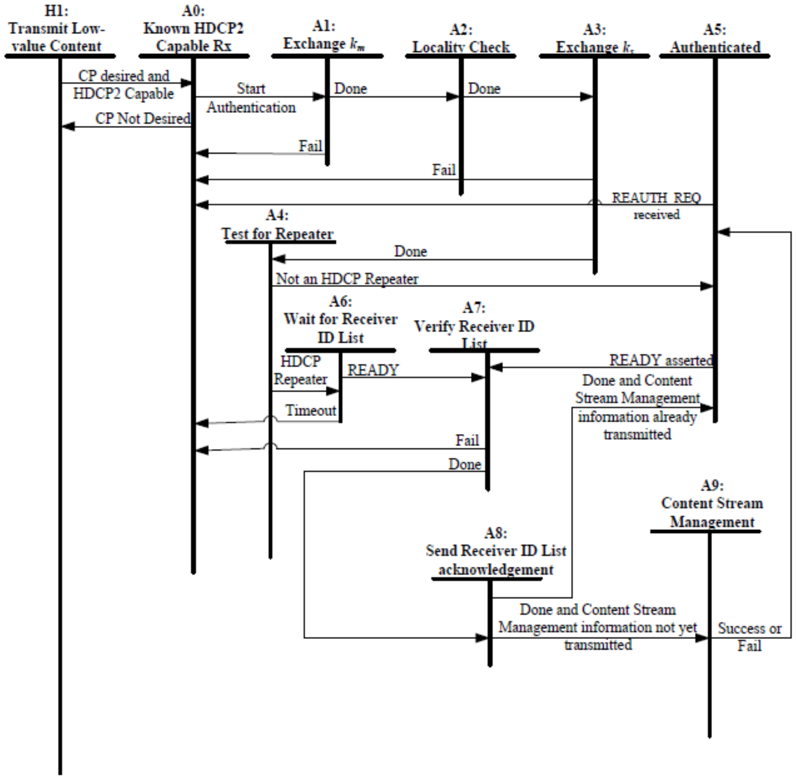
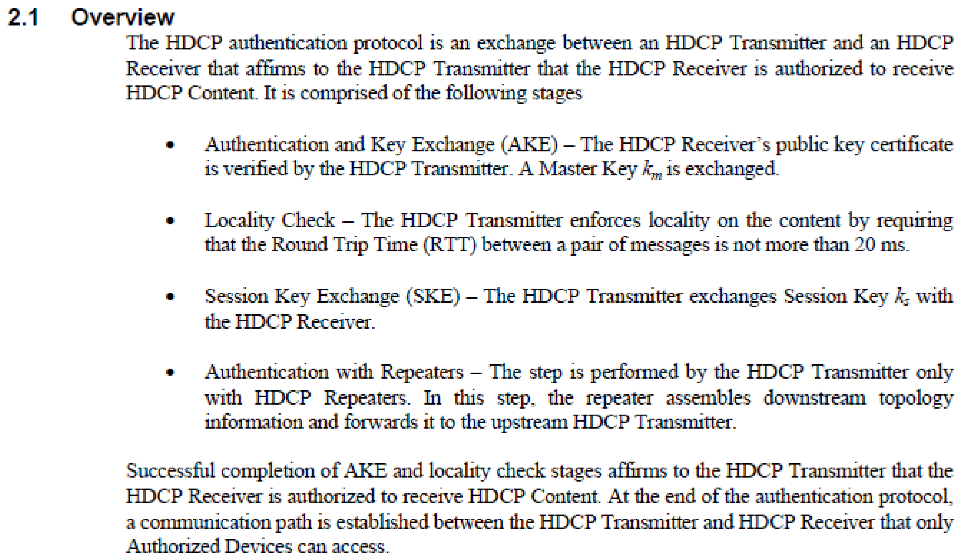


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27-30.

The Accused Product implements the HDCP 2.2 protocol to affirm that a second device is authorized to receive protected content.



2.1 Overview

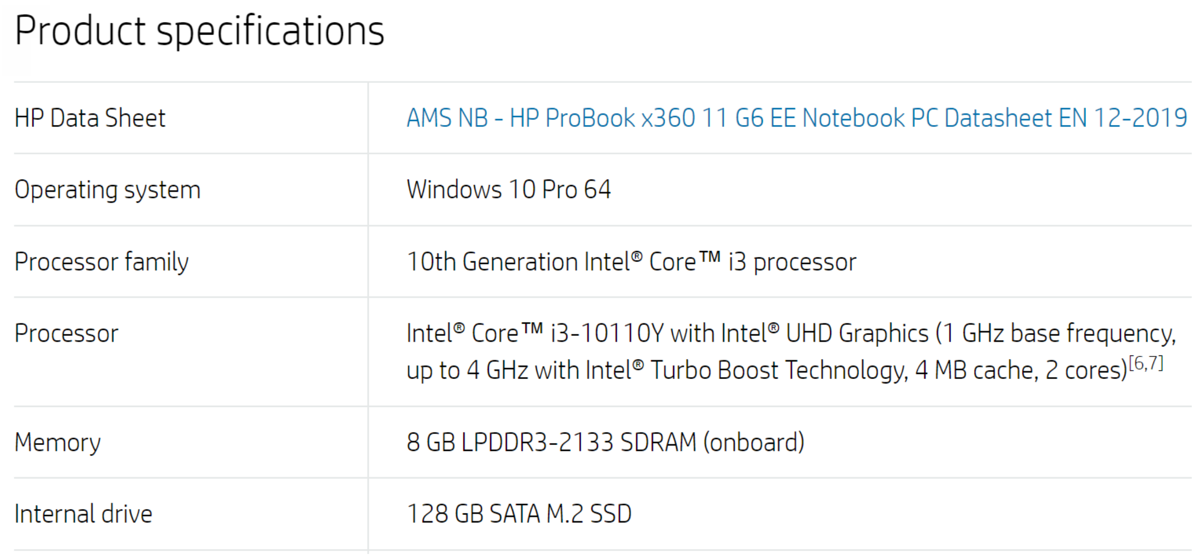
The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages

- Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged.
- Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms.
- Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver.
- Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter.

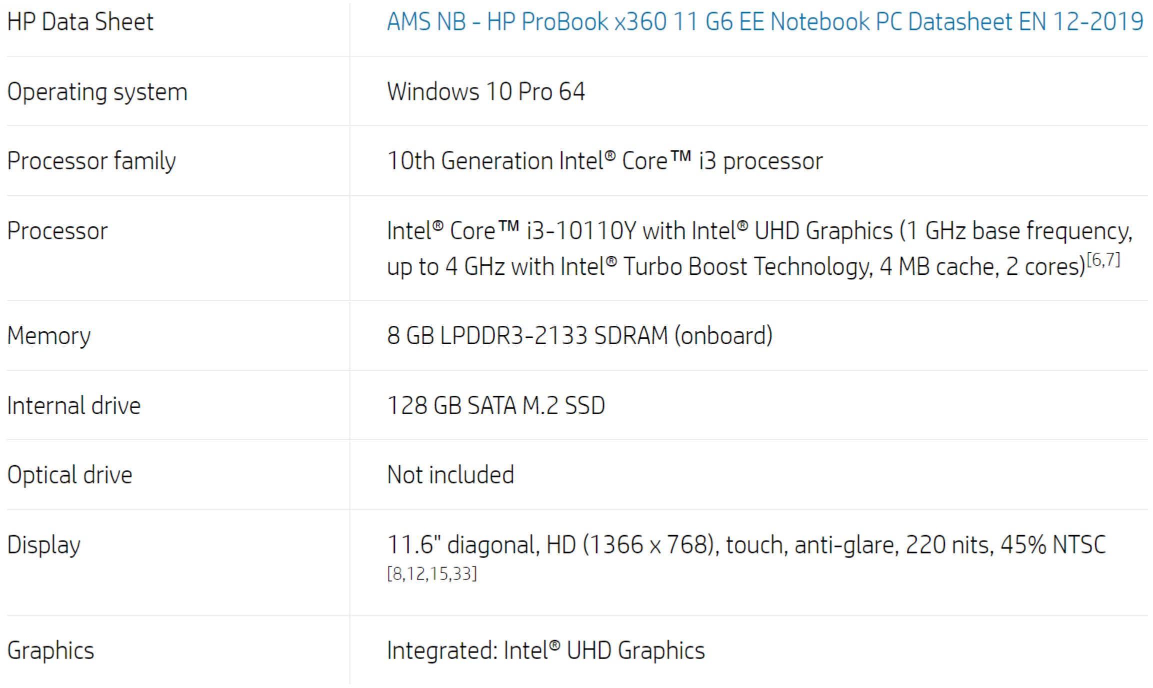
Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access.

Id. at 11.

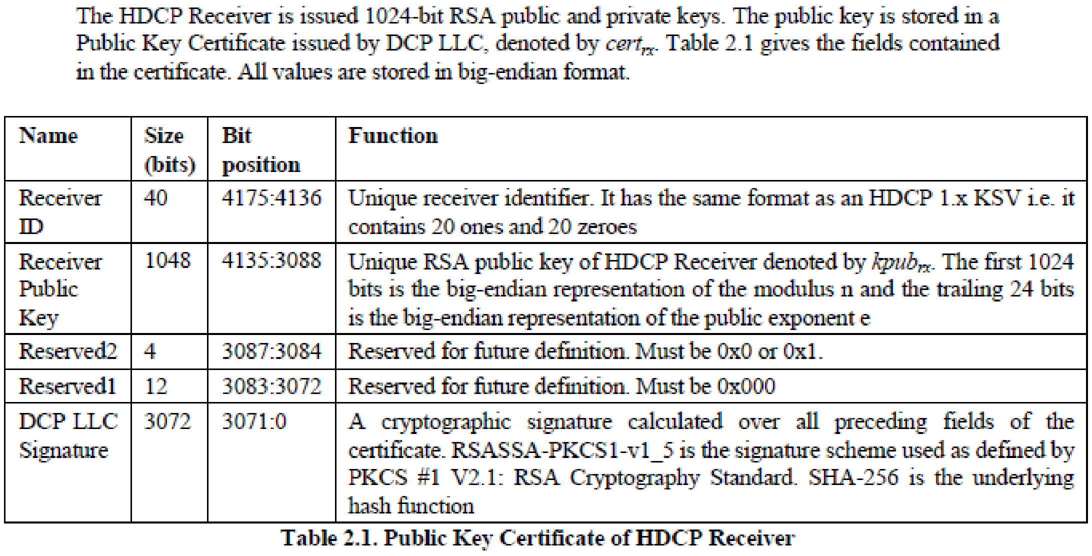
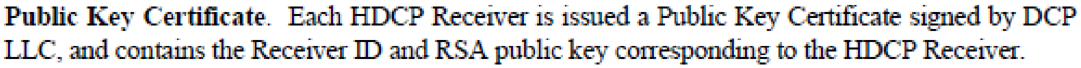
"a memory;"

<table>
<tr>
<td>a memory;</td>
<td>
<p>The Accused Product includes a memory.</p>
<p>For example, the Intel Processor includes a 4MB cache and the Accused Product also includes a 8GB onboard LPDDR3 memory in addition to a 128GB solid state hard drive.</p>

<p>Product specifications</p>
<table>
<tr><td>HP Data Sheet</td><td>AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019</td></tr>
<tr><td>Operating system</td><td>Windows 10 Pro 64</td></tr>
<tr><td>Processor family</td><td>10th Generation Intel® Core™ i3 processor</td></tr>
<tr><td>Processor</td><td>Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]</td></tr>
<tr><td>Memory</td><td>8 GB LPDDR3-2133 SDRAM (onboard)</td></tr>
<tr><td>Internal drive</td><td>128 GB SATA M.2 SSD</td></tr>
</table>
<p>HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc.</p>
</td>
</tr>
</table>

"a processor, the processor arranged to:"

<table>
<tr>
<td>a processor, the processor arranged to:</td>
<td>
<p>The Accused Product includes a processor.</p>
<p>For example, the Accused Product includes the Intel Processor integrated with the Intel GPU.</p>

<table>
<tr><td>HP Data Sheet</td><td>AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019</td></tr>
<tr><td>Operating system</td><td>Windows 10 Pro 64</td></tr>
<tr><td>Processor family</td><td>10th Generation Intel® Core™ i3 processor</td></tr>
<tr><td>Processor</td><td>Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]</td></tr>
<tr><td>Memory</td><td>8 GB LPDDR3-2133 SDRAM (onboard)</td></tr>
<tr><td>Internal drive</td><td>128 GB SATA M.2 SSD</td></tr>
<tr><td>Optical drive</td><td>Not included</td></tr>
<tr><td>Display</td><td>11.6" diagonal, HD (1366 x 768), touch, anti-glare, 220 nits, 45% NTSC [8,12,15,33]</td></tr>
<tr><td>Graphics</td><td>Integrated: Intel® UHD Graphics</td></tr>
</table>
<p>HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. <i>See also</i> NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.</p>
</td>
</tr>
</table>

"receive a certificate from the second device prior to sending a first signal;"

<table>
<tr>
<td>receive a certificate from the second device prior to sending a first signal;</td>
<td>
<p>The processor of the Accused Product is arranged to receive a certificate of the second device, <i>e.g.</i>, $cert_{rx}$, as part of the Authentication and Key Exchange (AKE) stage of the HDCP 2.2 protocol and prior to sending a first signal, <i>e.g.</i>, r_n of the LC_Init message.</p>
<p>The certificate, $cert_{rx}$, includes a Receiver ID for the second device, Receiver Public Key for the second device, and a cryptographic signature, amongst other information.</p>

<p>The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.</p>
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
<p>Table 2.1. Public Key Certificate of HDCP Receiver</p>
<p>HDMI HDCP 2.2 at 11.</p>

<p>Public Key Certificate. Each HDCP Receiver is issued a Public Key Certificate signed by DCP LLC, and contains the Receiver ID and RSA public key corresponding to the HDCP Receiver.</p>
<p><i>Id.</i> at 8.</p>
<p>The Accused Product receives the certificate from the second device as part of the AKE stage, irrespective of whether the Accused Product has a Master Key k_m stored corresponding to the Receiver ID.</p>
</td>
</tr>
</table>

"receive a certificate from the second device prior to sending a first signal;"

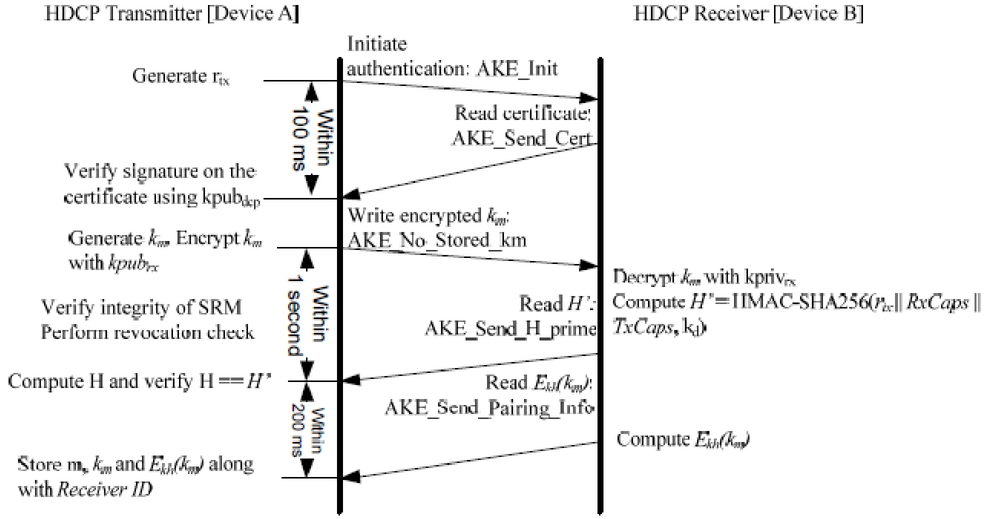


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

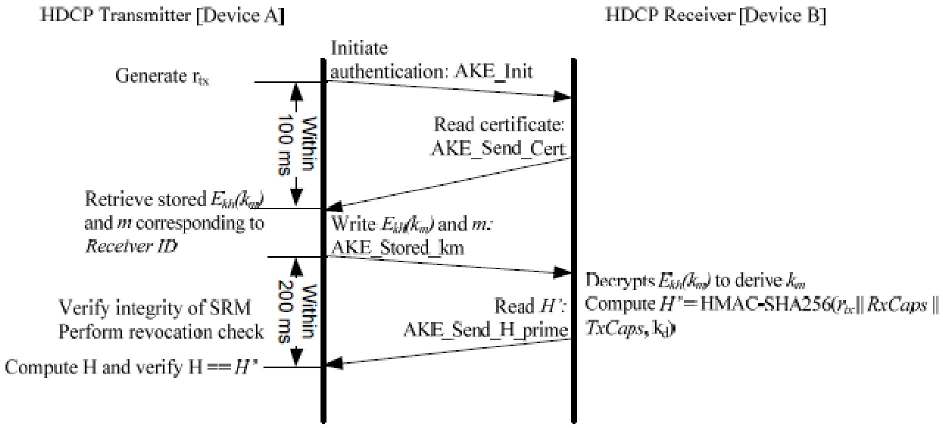
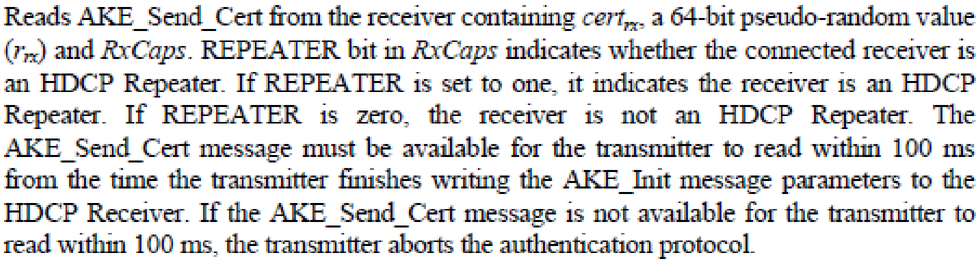


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

Id.

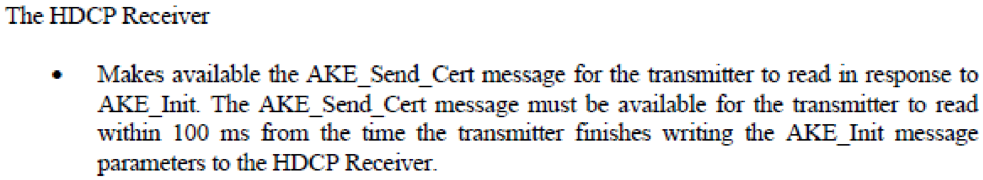
"receive a certificate from the second device prior to sending a first signal;"

The Accused Product receives the certificate from the second device as part of the AKE_Send_Cert message.



Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol.

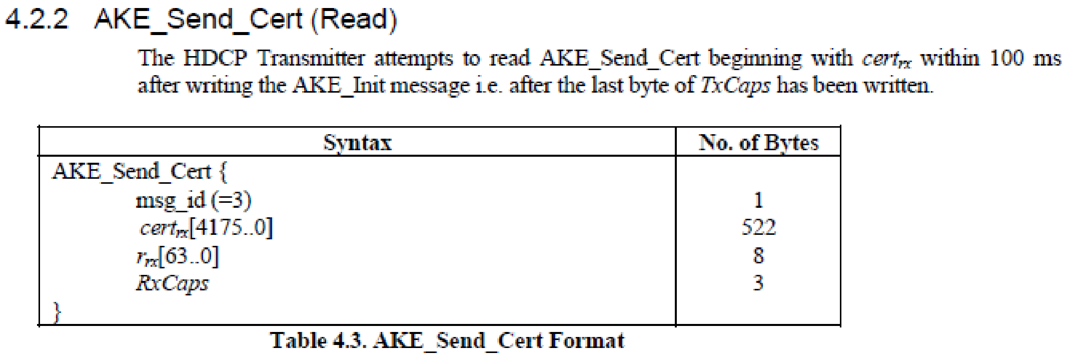
Id. at 13.



The HDCP Receiver

- Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

Id. at 14.



4.2.2 AKE_Send_Cert (Read)

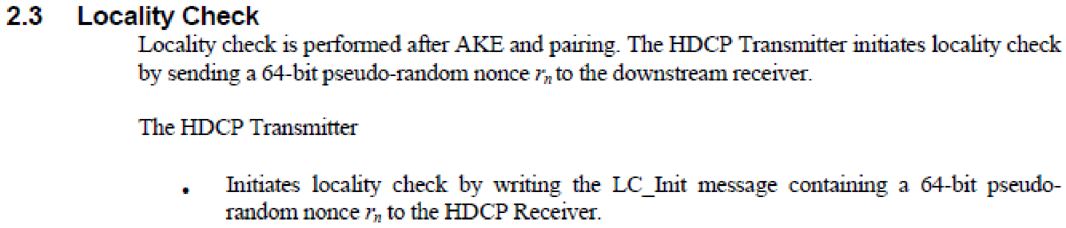
The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

The Accused Product receives the certificate from the second device during the AKE stage prior to sending a first signal, *e.g.*, r_n of the LC_Init message, as part of a Locality Check.

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter - Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. *Id.* at 16. *See also*:

U.S. Patent No. 9,436,809: Claim 49

"receive a certificate from the second device prior to sending a first signal;"

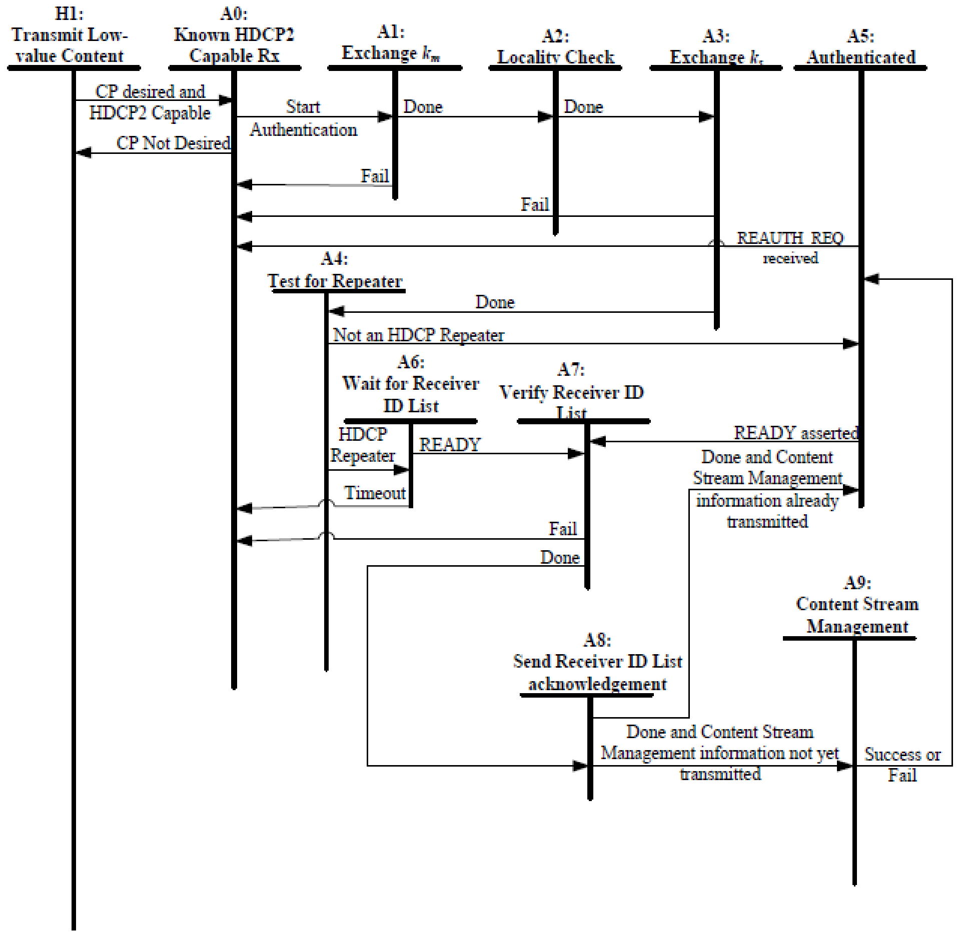
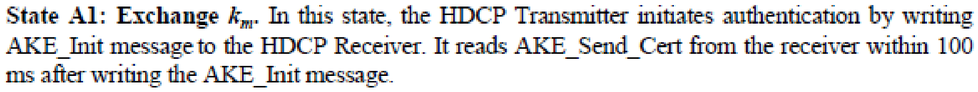


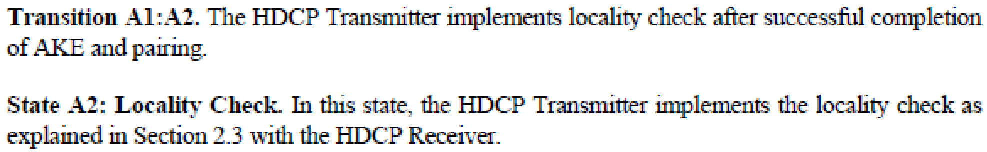
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.



State A1: Exchange k_m. In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message.

Id. at 28.

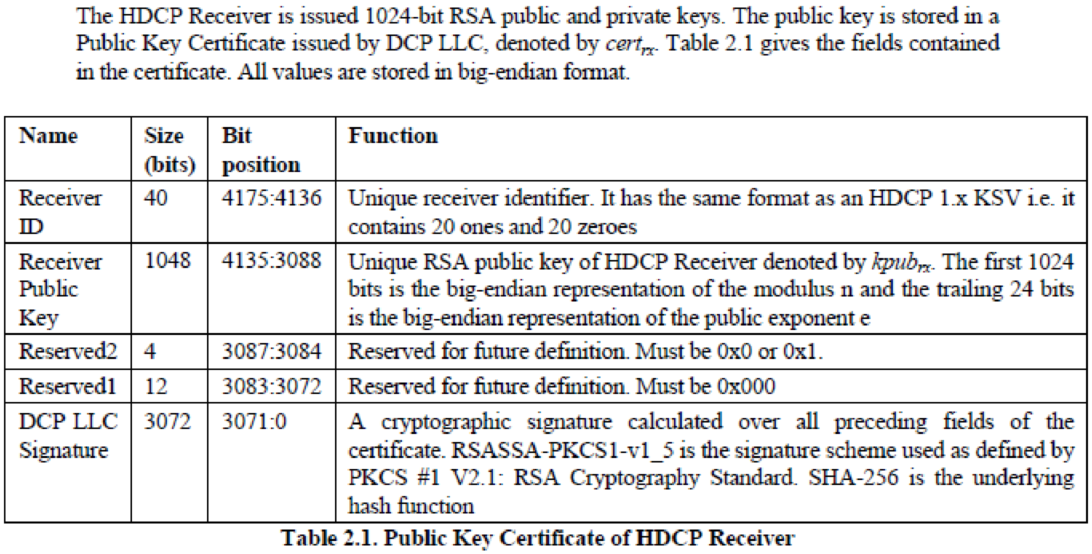
	 **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id.*

"determine from the certificate if the second device is compliant;"

<table>
<tr>
<td>determine from the certificate if the second device is compliant;</td>
<td>

The processor of the Accused Product is arranged to determine from the certificate whether the second device is compliant.

The Accused Product determines from the certificate, *e.g.*, $cert_{rx}$, and as part of the Authentication and Key Exchange (AKE) stage, whether the second device is compliant. For example, $cert_{rx}$ includes a Receiver ID, Receiver Public Key, and a cryptographic signature.



The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

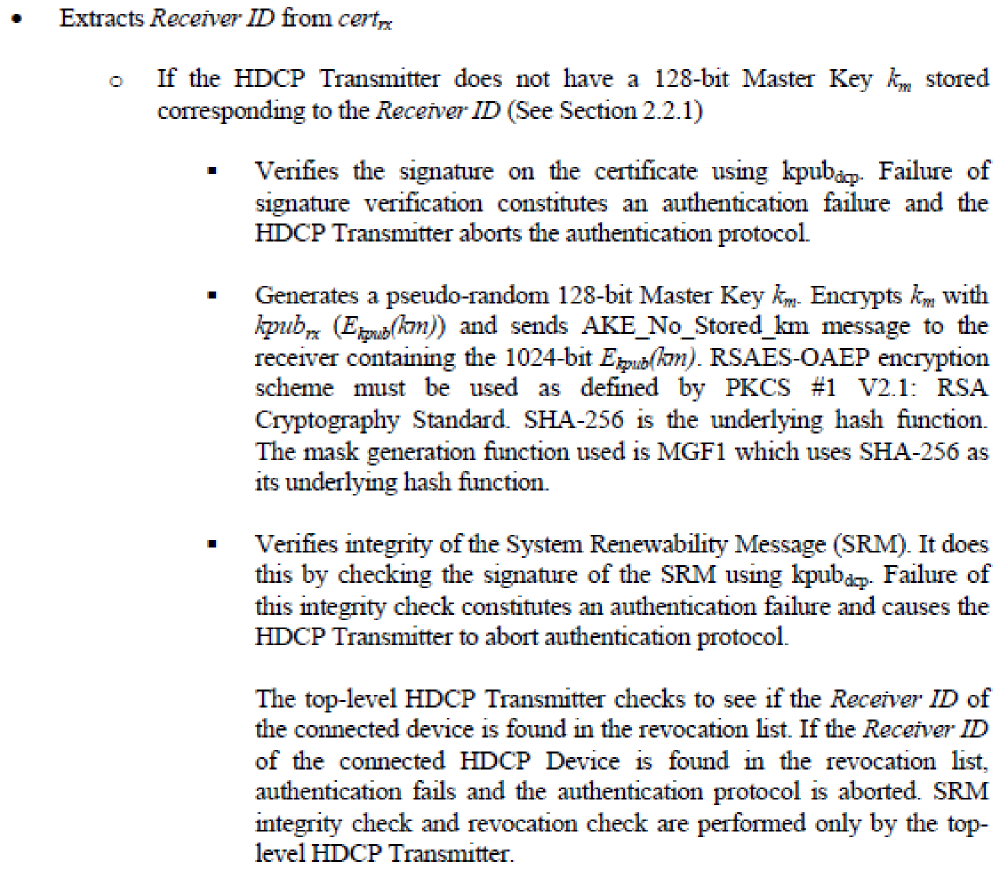
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>

Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.

The Accused Product determines, for example, whether the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature.

</td>
</tr>
</table>



- Extracts *Receiver ID* from $cert_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using $kpub_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with the compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2, *available at* https://digital-cp.com/sites/default/files/HDCP%20License%20Agreement_March%206%2C%202017_FOR%20REVIEW%20ONLY.pdf.

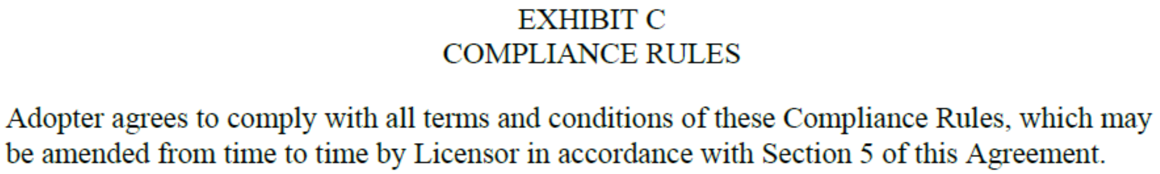


EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

Id. at Exhibit C.

See also:

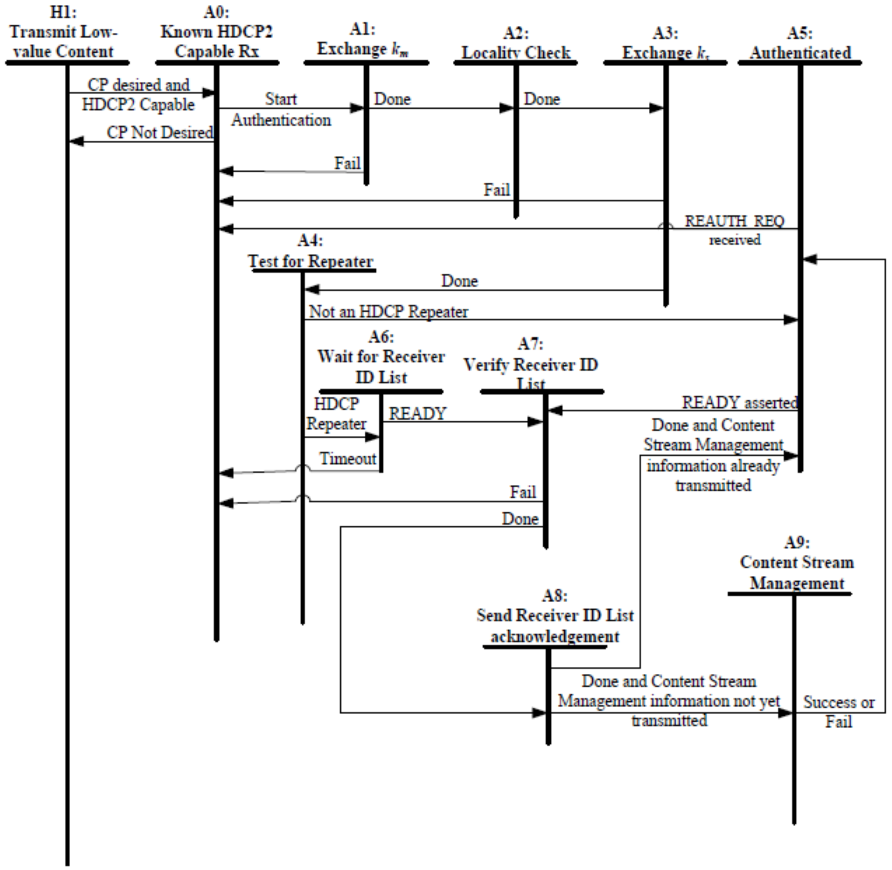
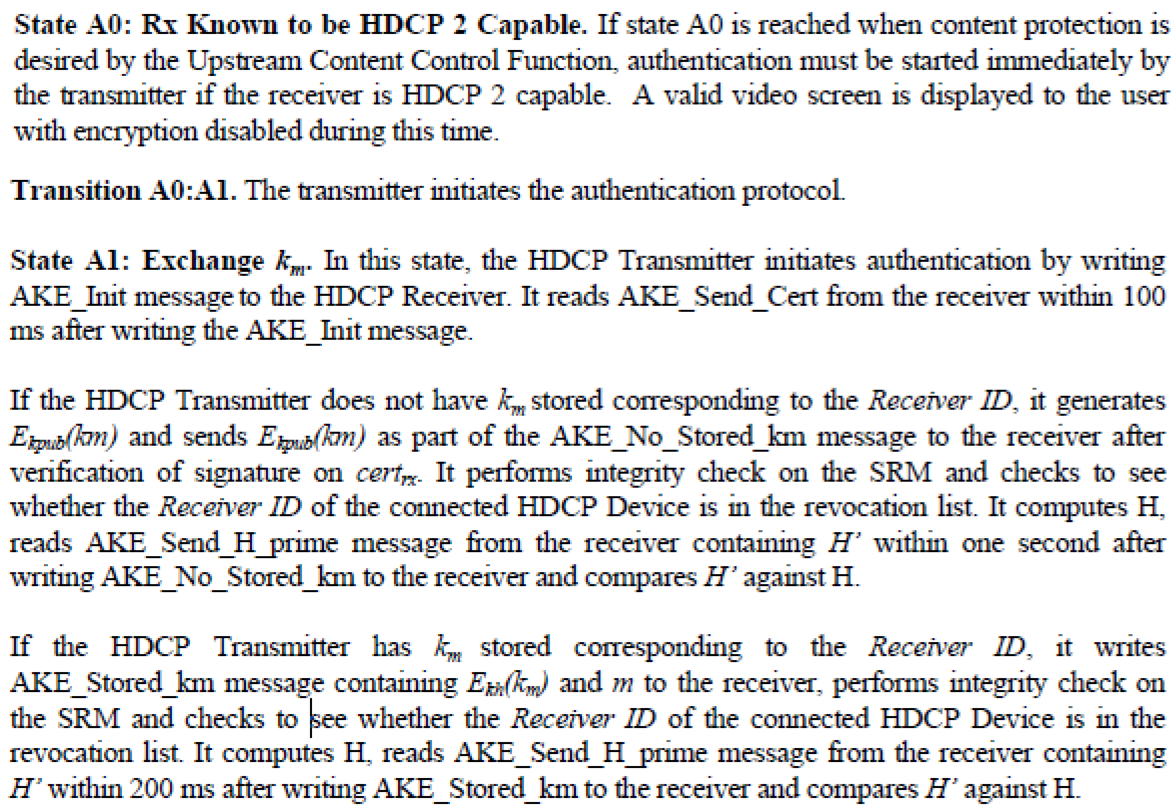


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

HDMI HDCP 2.2 at 27.



State A0: Rx Known to be HDCP 2 Capable. If state A0 is reached when content protection is desired by the Upstream Content Control Function, authentication must be started immediately by the transmitter if the receiver is HDCP 2 capable. A valid video screen is displayed to the user with encryption disabled during this time.

Transition A0:A1. The transmitter initiates the authentication protocol.

State A1: Exchange k_m. In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message.

If the HDCP Transmitter does not have k_m stored corresponding to the *Receiver ID*, it generates $E_{kpub}(km)$ and sends $E_{kpub}(km)$ as part of the AKE_No_Stored_km message to the receiver after verification of signature on $cert_{rx}$. It performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within one second after writing AKE_No_Stored_km to the receiver and compares *H'* against H.

If the HDCP Transmitter has k_m stored corresponding to the *Receiver ID*, it writes AKE_Stored_km message containing $E_{kh}(k_m)$ and *m* to the receiver, performs integrity check on the SRM and checks to see whether the *Receiver ID* of the connected HDCP Device is in the revocation list. It computes H, reads AKE_Send_H_prime message from the receiver containing *H'* within 200 ms after writing AKE_Stored_km to the receiver and compares *H'* against H.

Id. at 28.

"provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;"

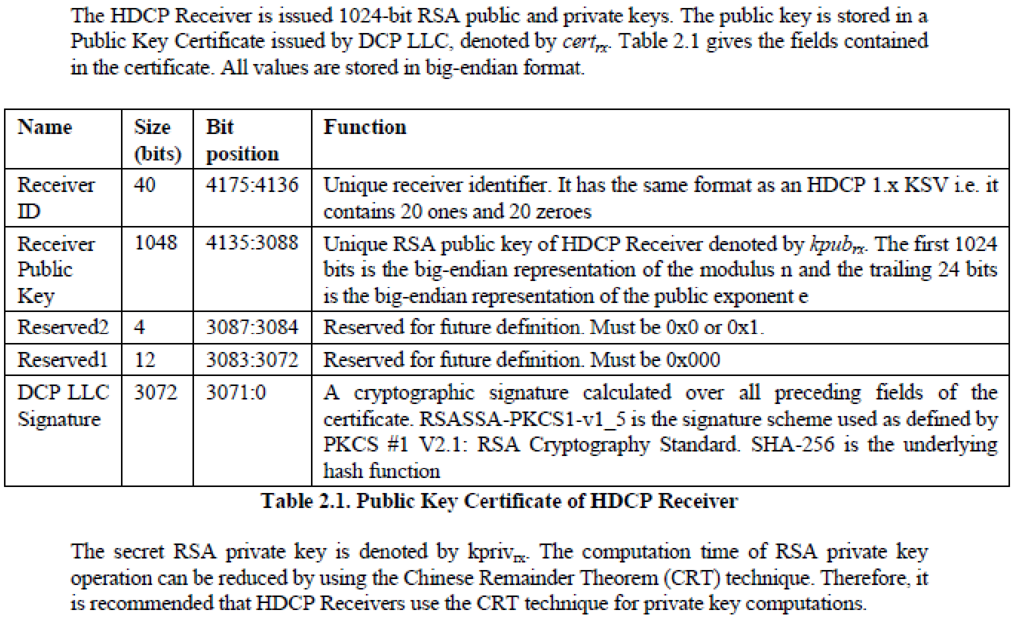
<table>
<tr>
<td>provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;</td>
<td>

The processor of the Accused Product is arranged to provide a secret, *e.g.*, k_m, to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number.

The second device has a public key, *e.g.*, $kpub_{rx}$, and a private key, *e.g.*, $kpriv_{rx}$, wherein the public key and private key are a pair. The public key is stored in the certificate, $cert_{rx}$.

Device Key Set. An HDCP Receiver has a Device Key Set, which consists of its corresponding Device Secret Keys along with the associated Public Key Certificate.

HDMI HDCP 2.2 at 6.



The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>

Table 2.1. Public Key Certificate of HDCP Receiver

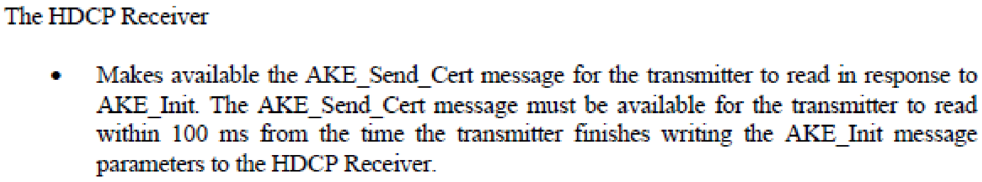
The secret RSA private key is denoted by $kpriv_{rx}$. The computation time of RSA private key operation can be reduced by using the Chinese Remainder Theorem (CRT) technique. Therefore, it is recommended that HDCP Receivers use the CRT technique for private key computations.

Id. at 11.

The Accused Product receives the public key of the second device, *e.g.*, $kpub_{rx}$, as part of the AKE_Send_Cert message.

</td>
</tr>
</table>

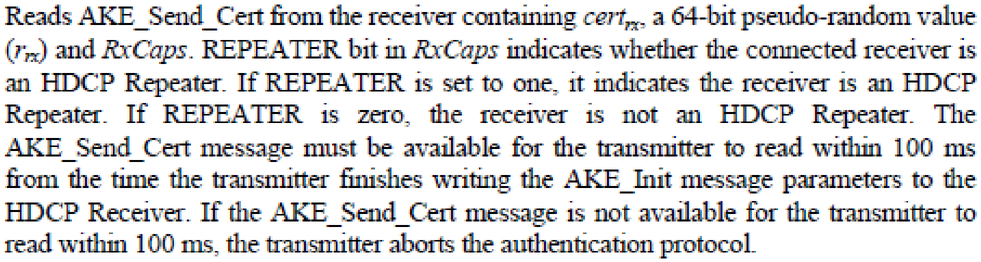
"provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;"



The HDCP Receiver

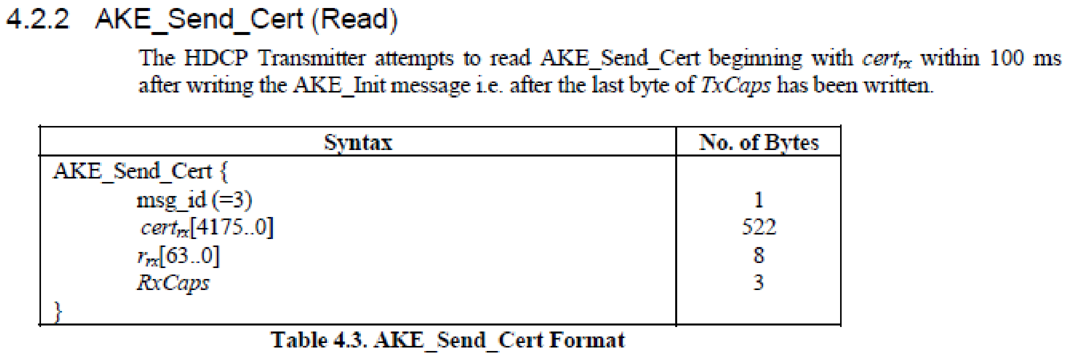
- Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

Id. at 14.



Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol.

Id. at 13.



4.2.2 AKE_Send_Cert (Read)

The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

The Accused Product provides the secret, *e.g.*, k_m, to the second device via encryption using the second device's public key, *e.g.*, $kpub_{rx}$, if the second device is compliant, *e.g.*, as determined based on $cert_{rx}$ received by the Accused Product as part of the AKE_Send_Cert message.

U.S. Patent No. 9,436,809: Claim 49

"provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;"

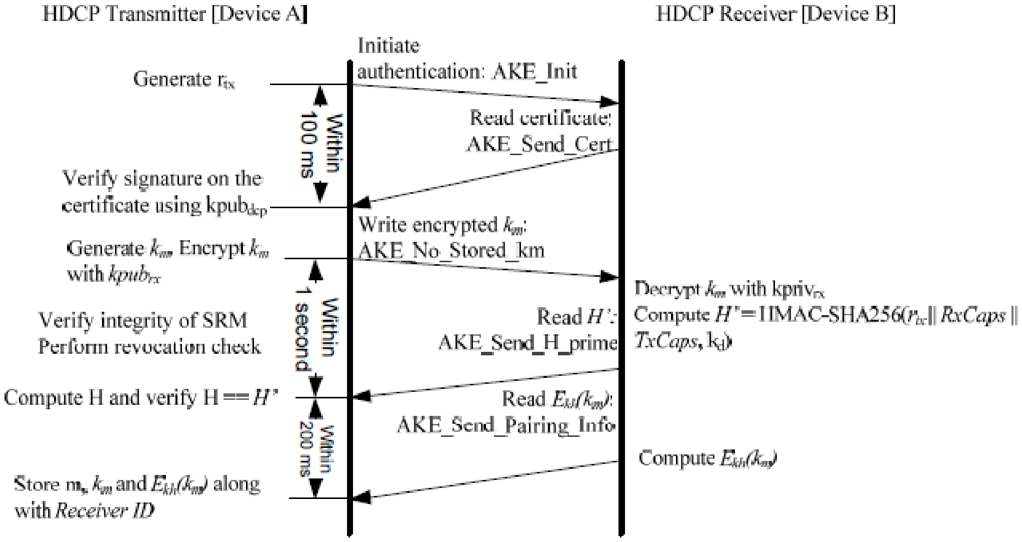
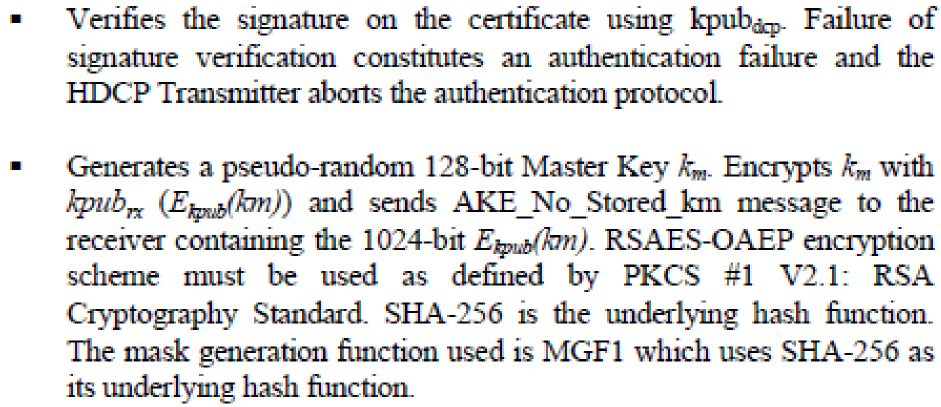


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

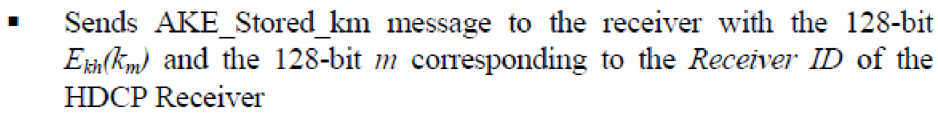
Id. at 12.



- Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.

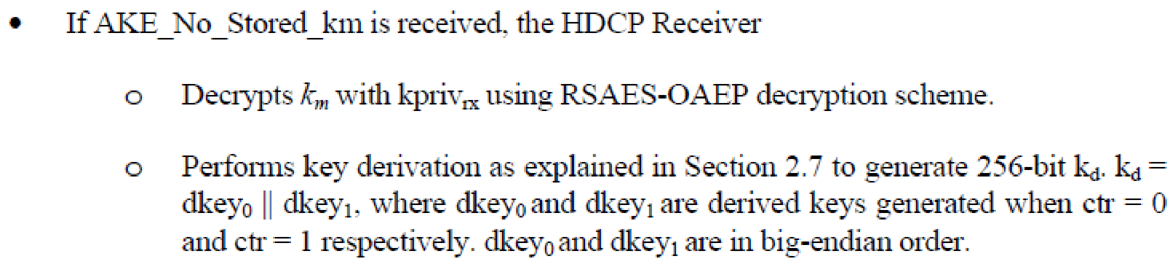
- Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.

Id. at 13.



- Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit m corresponding to the *Receiver ID* of the HDCP Receiver

Id. at 14.



- If AKE_No_Stored_km is received, the HDCP Receiver
  - Decrypts k_m with kpriv_{rx} using RSAES-OAEP decryption scheme.
  - Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. $k_d = \text{dkey}_0 \,||\, \text{dkey}_1$, where dkey_0 and dkey_1 are derived keys generated when ctr = 0 and ctr = 1 respectively. dkey_0 and dkey_1 are in big-endian order.

Id.

For example, a valid signature in the certificate indicates that the second device is compliant with the compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2.

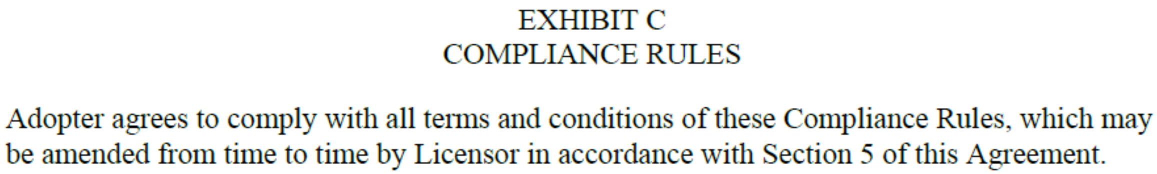
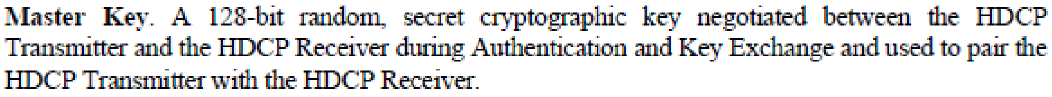


EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

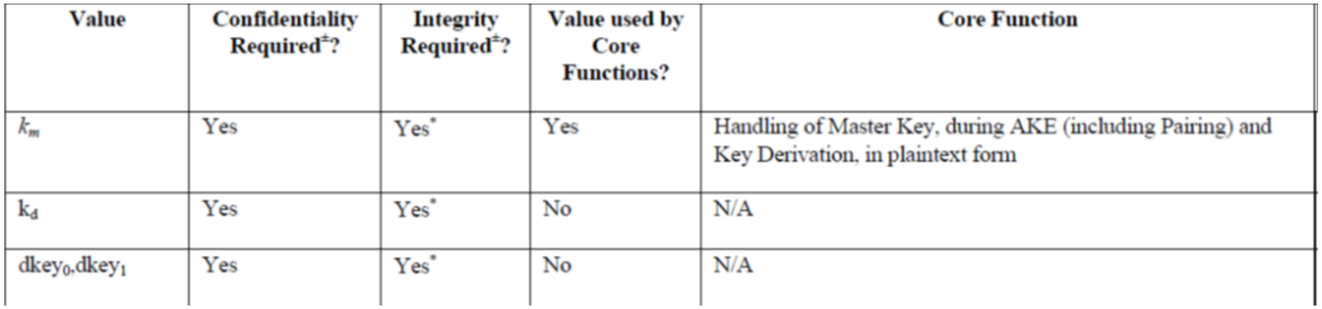
Id. at Exhibit C.

k_m, the Master Key, is a secret and comprises a random number.



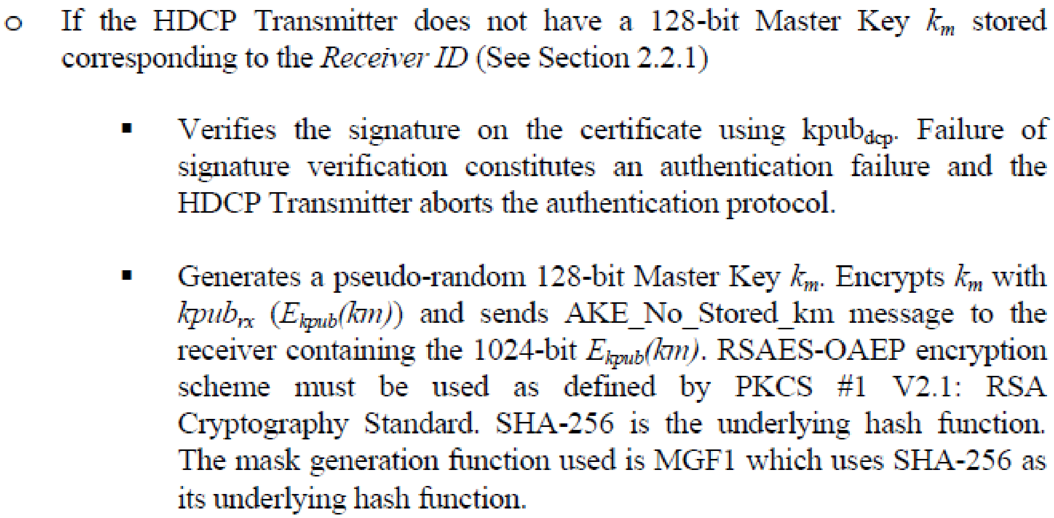
Master Key. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver.

HDMI HDCP 2.2 at 8.



Value	Confidentiality Required[±]?	Integrity Required[±]?	Value used by Core Functions?	Core Function
k_m	Yes	Yes[*]	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes[*]	No	N/A
$dkey_0$,$dkey_1$	Yes	Yes[*]	No	N/A

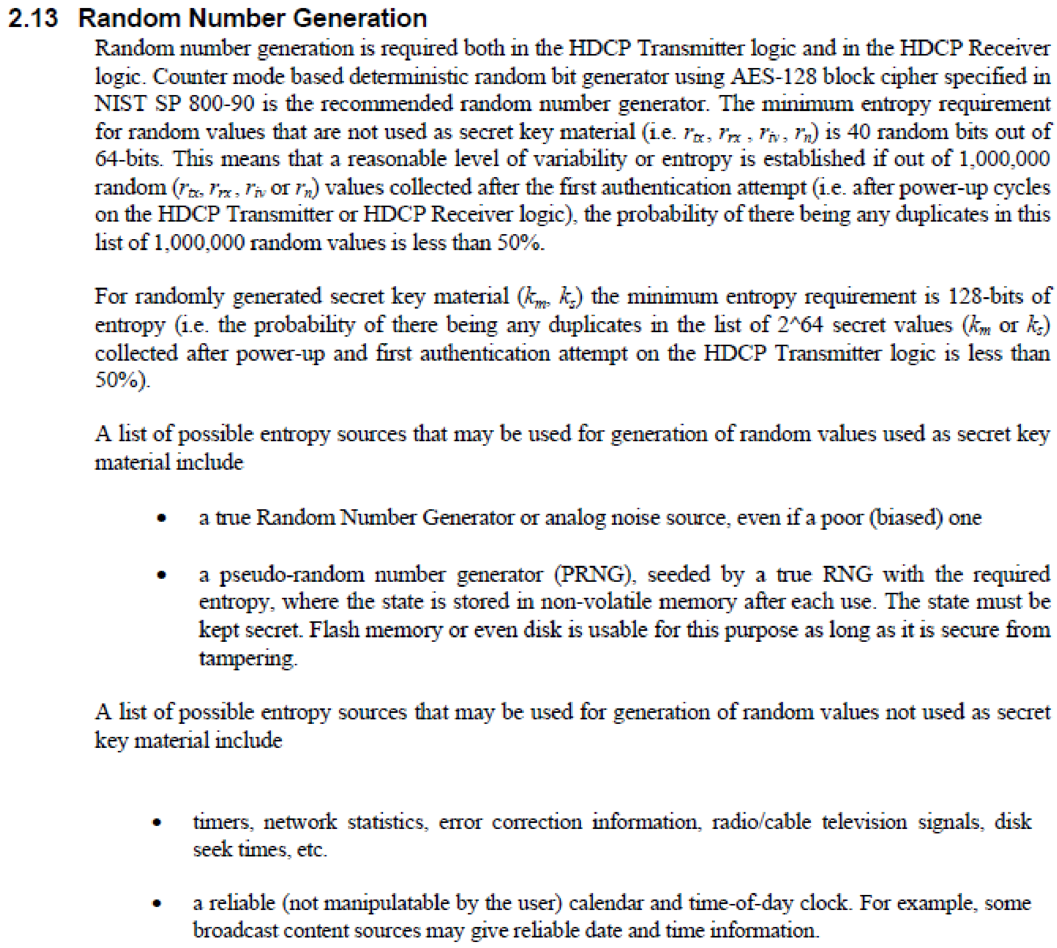
Id. at 67 (abridged).



- If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using $kpub_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.

Id. at 13.

"provide a secret to the second device via encryption by a public key of a private/public key-pair of the second device, if the second device is compliant, said secret comprising a random number;"

<table>
<tr>
<td></td>
<td>


2.13 Random Number Generation

Random number generation is required both in the HDCP Transmitter logic and in the HDCP Receiver logic. Counter mode based deterministic random bit generator using AES-128 block cipher specified in NIST SP 800-90 is the recommended random number generator. The minimum entropy requirement for random values that are not used as secret key material (i.e. r_{tx}, r_{rx}, r_{iv}, r_n) is 40 random bits out of 64-bits. This means that a reasonable level of variability or entropy is established if out of 1,000,000 random (r_{tx}, r_{rx}, r_{iv} or r_n) values collected after the first authentication attempt (i.e. after power-up cycles on the HDCP Transmitter or HDCP Receiver logic), the probability of there being any duplicates in this list of 1,000,000 random values is less than 50%.

For randomly generated secret key material (k_m, k_s) the minimum entropy requirement is 128-bits of entropy (i.e. the probability of there being any duplicates in the list of 2^64 secret values (k_m or k_s) collected after power-up and first authentication attempt on the HDCP Transmitter logic is less than 50%).

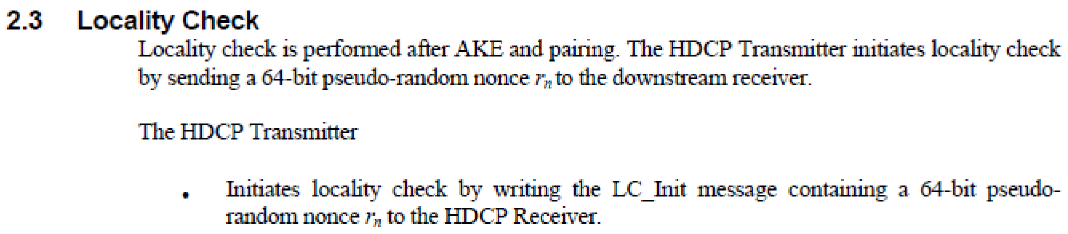
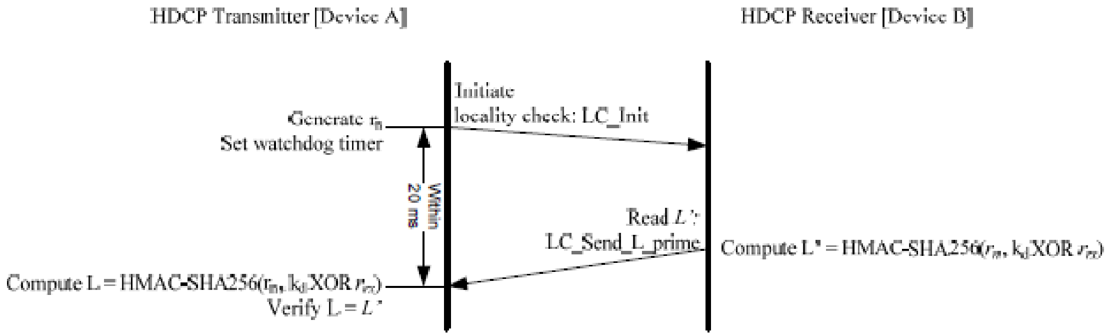
A list of possible entropy sources that may be used for generation of random values used as secret key material include

- a true Random Number Generator or analog noise source, even if a poor (biased) one
- a pseudo-random number generator (PRNG), seeded by a true RNG with the required entropy, where the state is stored in non-volatile memory after each use. The state must be kept secret. Flash memory or even disk is usable for this purpose as long as it is secure from tampering.

A list of possible entropy sources that may be used for generation of random values not used as secret key material include

- timers, network statistics, error correction information, radio/cable television signals, disk seek times, etc.
- a reliable (not manipulatable by the user) calendar and time-of-day clock. For example, some broadcast content sources may give reliable date and time information.

Id. at 45-46.
</td>
</tr>
</table>

provide the first signal to the second device;	The processor of the Accused Product is arranged to provide the first signal, *e.g.*, r_n of the LC_Init message to the second device.  **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. HDMI HDCP 2.2 at 16.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** *Id*. at 17. *See also*:

"provide the first signal to the second device;"

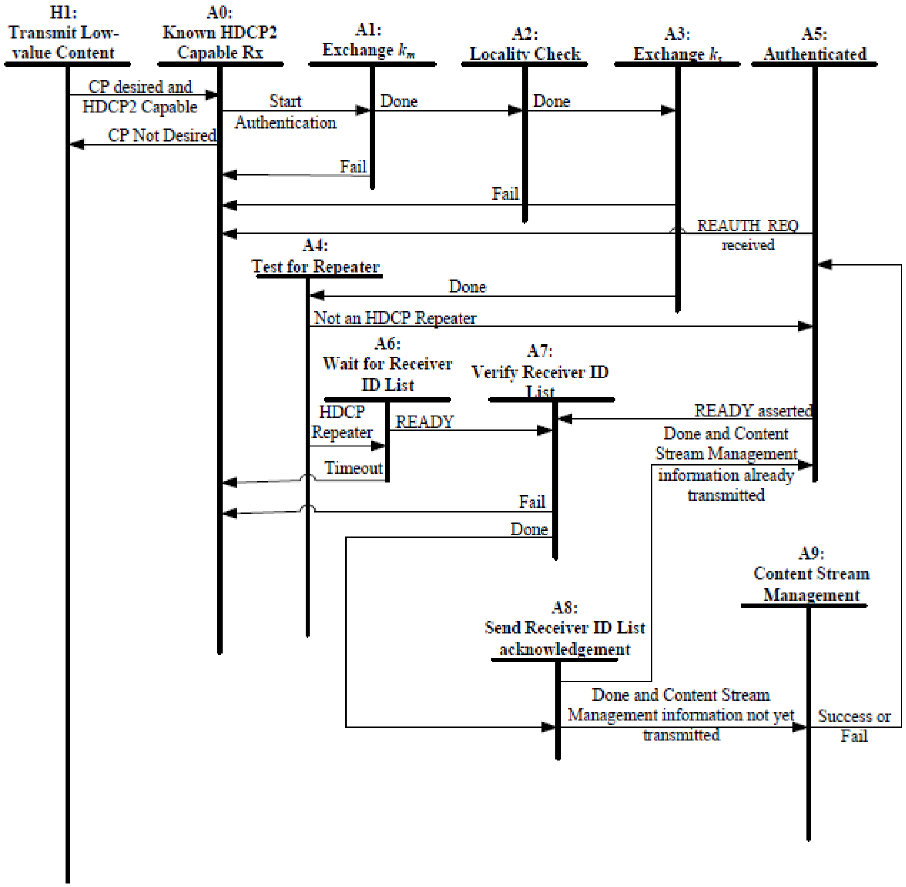
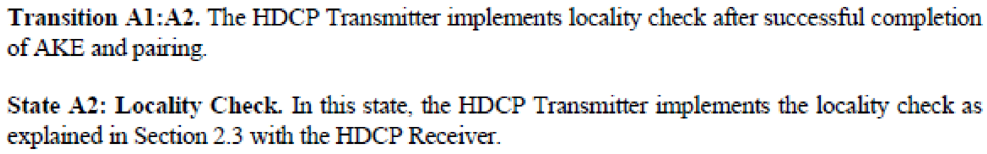


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

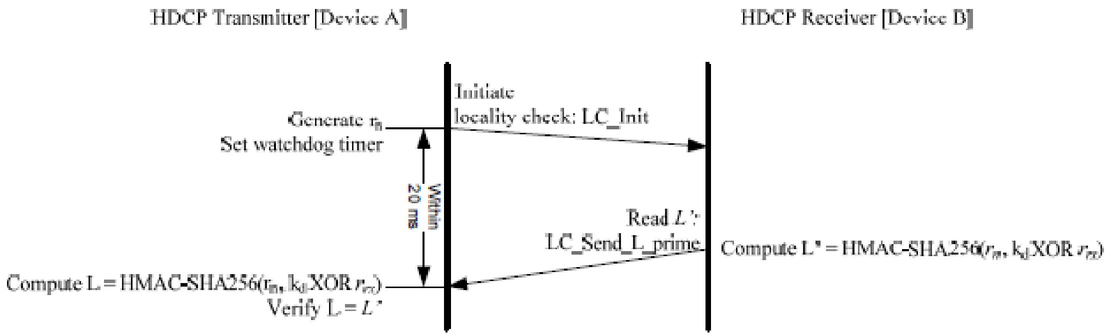
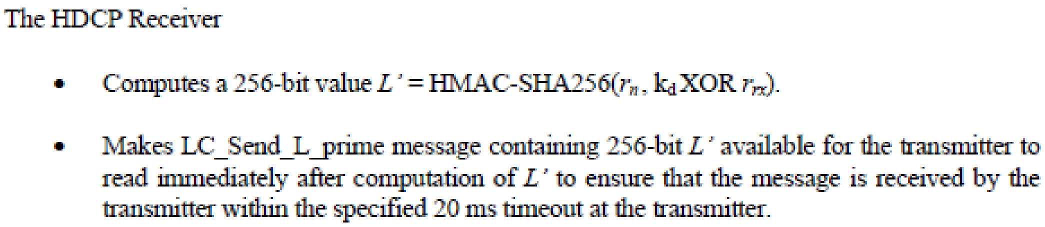


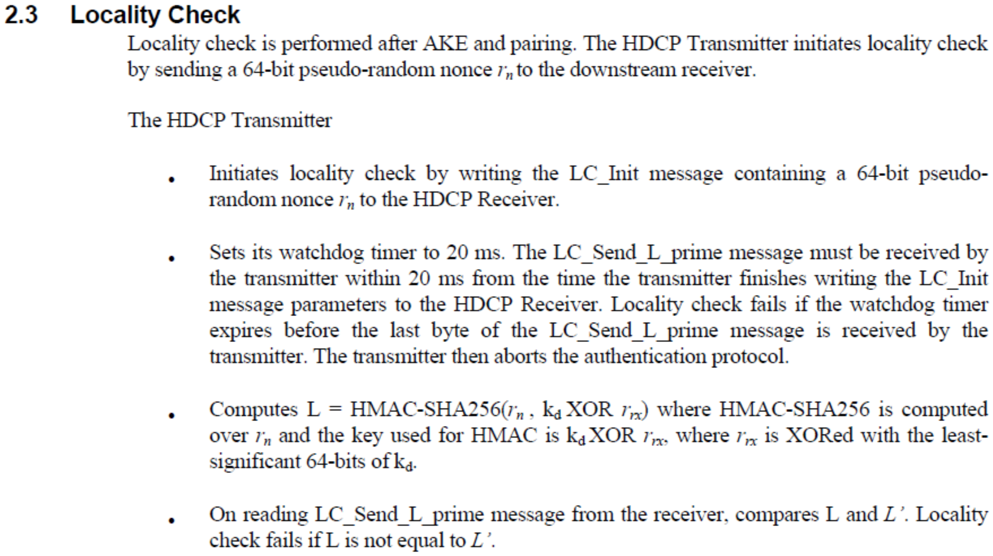
Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"receive a second signal from the second device after providing the first signal;"

receive a second signal from the second device after providing the first signal;	The processor of the Accused Product is arranged to receive a second signal, *e.g.*, the LC_Send_L_prime message including L', from the second device after providing the first signal, *e.g.*, r_n of the LC_Init message.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** HDMI HDCP 2.2 at 17.  The HDCP Receiver • Computes a 256-bit value L' = HMAC-SHA256(r_n, k_d XOR r_{rx}). • Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. *Id.*



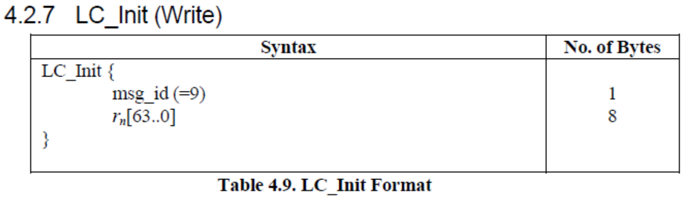
2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes $L = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$ where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

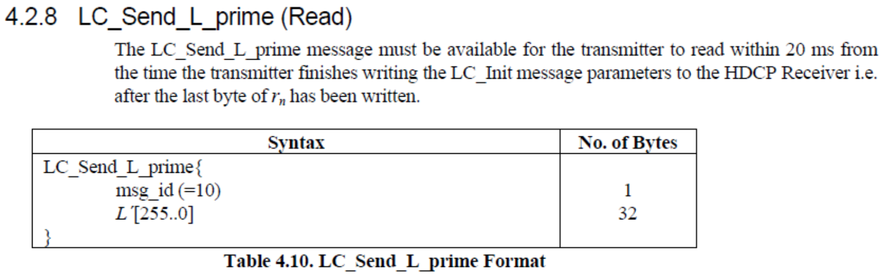


4.2.7 LC_Init (Write)

Syntax	No. of Bytes
LC_Init {	
msg_id (=9)	1
r_n[63..0]	8
}	

Table 4.9. LC_Init Format

Id. at 59.



4.2.8 LC_Send_L_prime (Read)

The LC_Send_L_prime message must be available for the transmitter to read within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver i.e. after the last byte of r_n has been written.

Syntax	No. of Bytes
LC_Send_L_prime{	
msg_id (=10)	1
L'[255..0]	32
}	

Table 4.10. LC_Send_L_prime Format

Id.

See also:

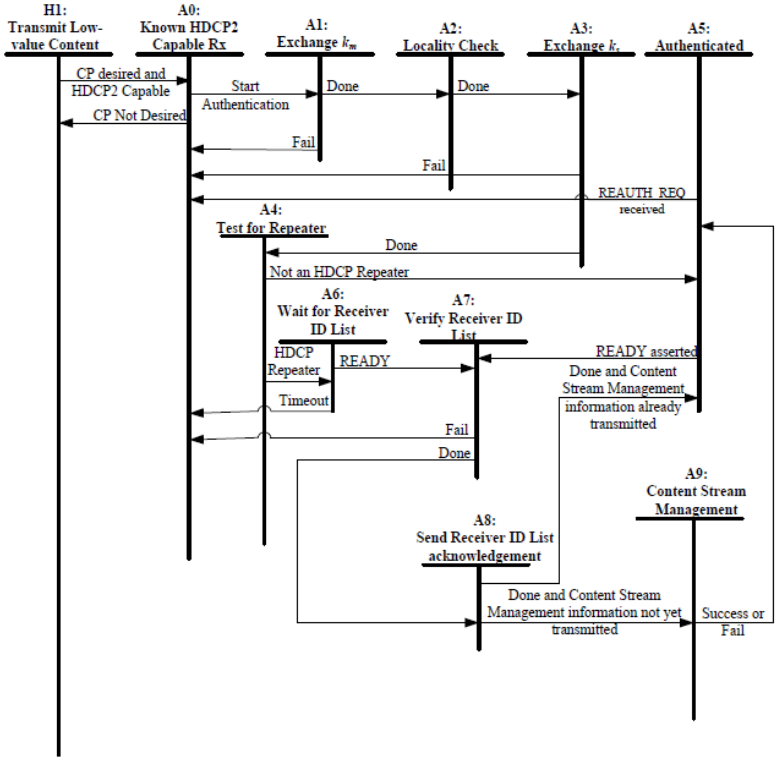
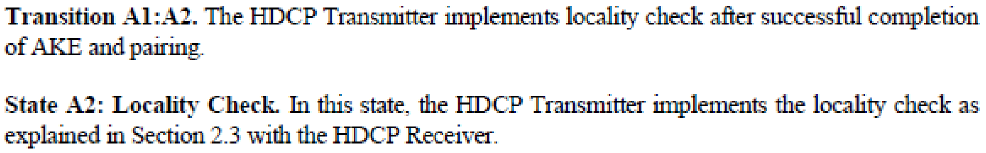


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

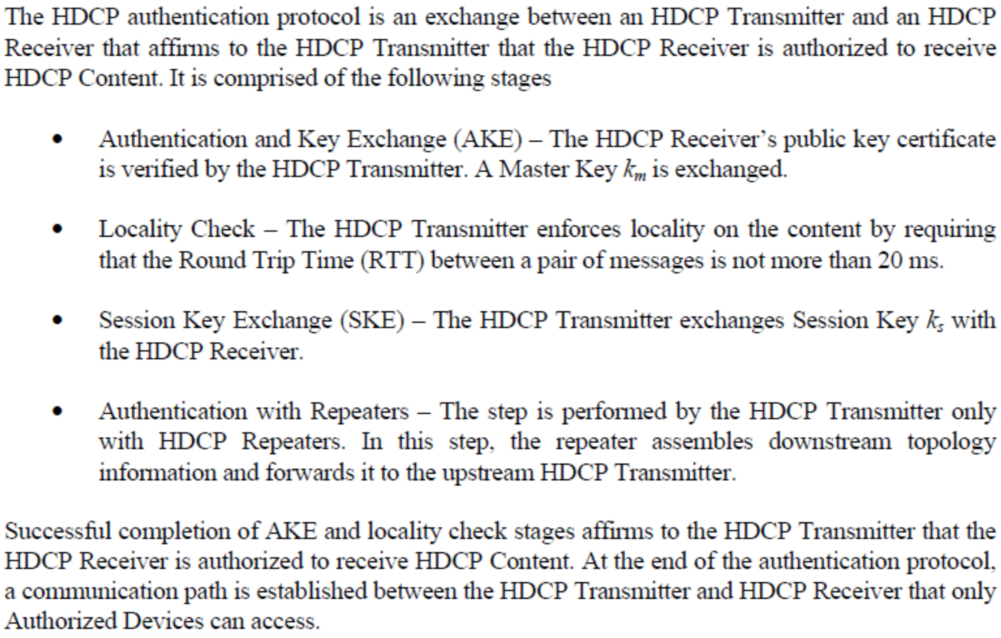


Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

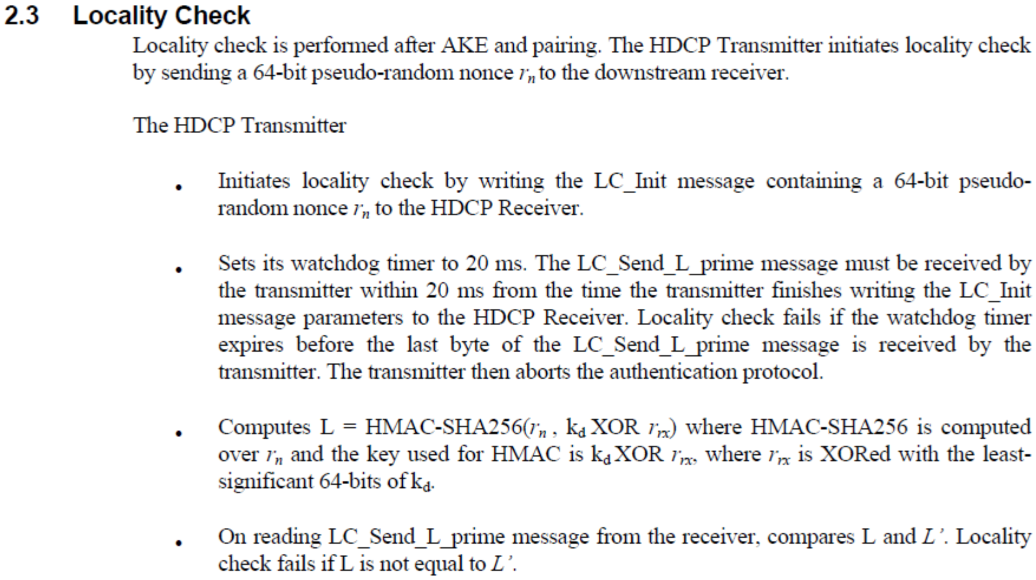
Id. at 28.

"determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;"

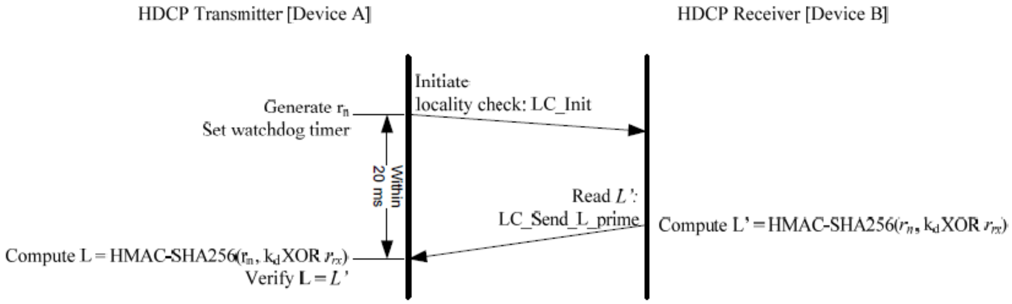
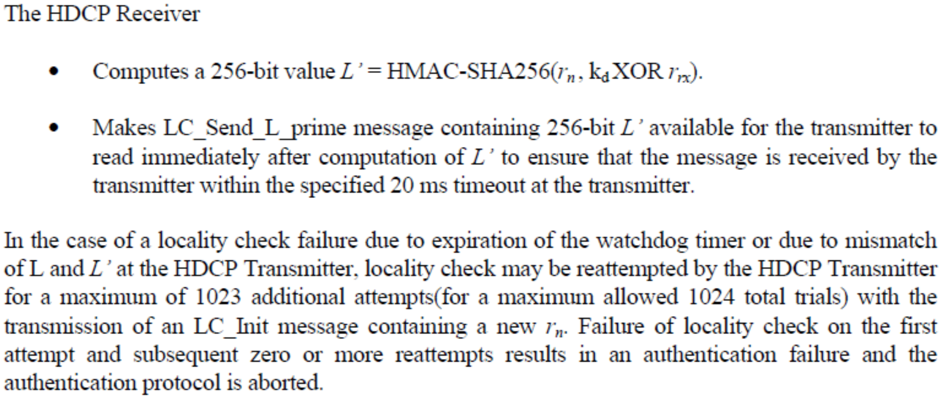
determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;	The processor of the Accused Product is arranged to determine whether the second signal, *e.g.*, L', is derived from the secret, *e.g.*, k_m, by determining whether the second signal is the first signal, *e.g.*, r_n, modified based on the secret. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Locality Check requires the Accused Product (transmitter) to determine that L' received via the LC_Send_L_prime message is derived from the secret by matching L' to value L which is the first signal, *e.g.*, r_n, modified based on the secret (*e.g.*, L is computed based on r_n and k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m).

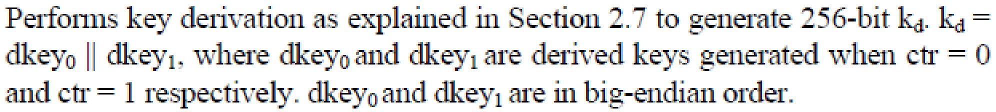
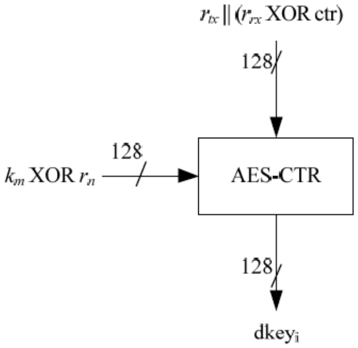
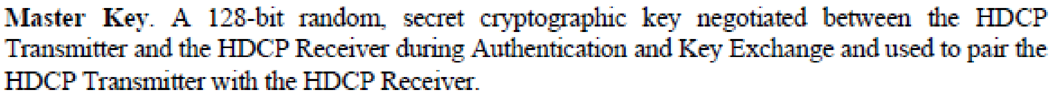
U.S. Patent No. 9,436,809: Claim 49

"determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;"

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter - Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. - Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. - Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. - On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

"determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;"

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver**  The HDCP Receiver - Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$. - Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted. *Id*. at 17. The second signal, *e.g.*, L', is derived from a secret. The value of L' is derived from k_d.  Compute L' = HMAC-SHA256(r_n, k_dXOR r_{rx}) *Id*. The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

	 Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order. *Id*. at 14-15.  **Figure 2.10. Key Derivation** *Id*. at 25.  **Master Key**. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver. *Id*. at 8. *See also*:

"determine if the second signal is derived from the secret by determining whether the second signal is the first signal modified based on the secret;"

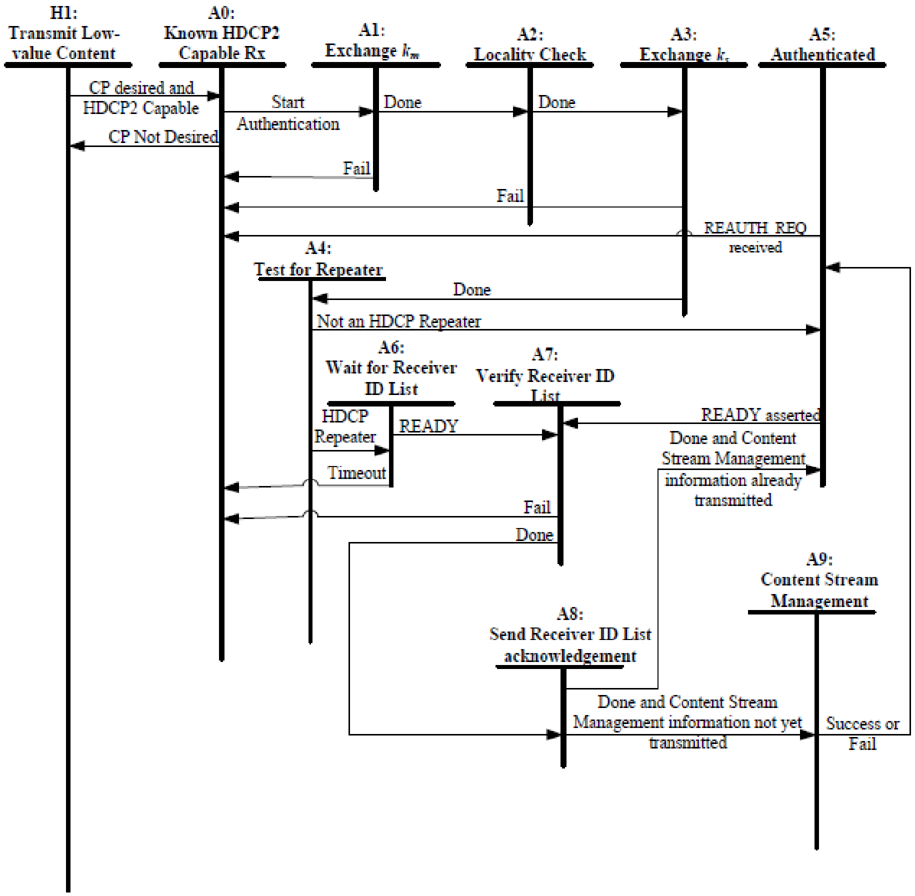


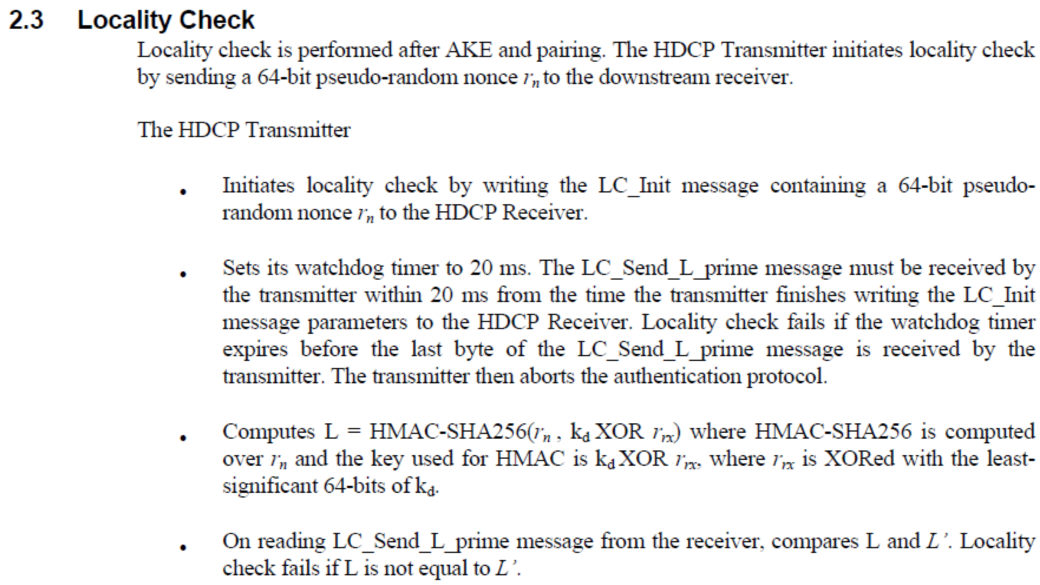
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

> **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing.
>
> **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and"

determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and	The processor of the Accused Product is arranged to determine whether a time difference between providing the first signal, *e.g.*, the LC_Init message including r_n, and receiving the second signal, *e.g.*, the LC_Send_L_prime message including L', is less than a predetermined time. The Locality Check requires the Accused Product to determine that the time between the providing of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than a predetermined time of 20 ms.  **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

<table>
<tr>
<td></td>
<td>

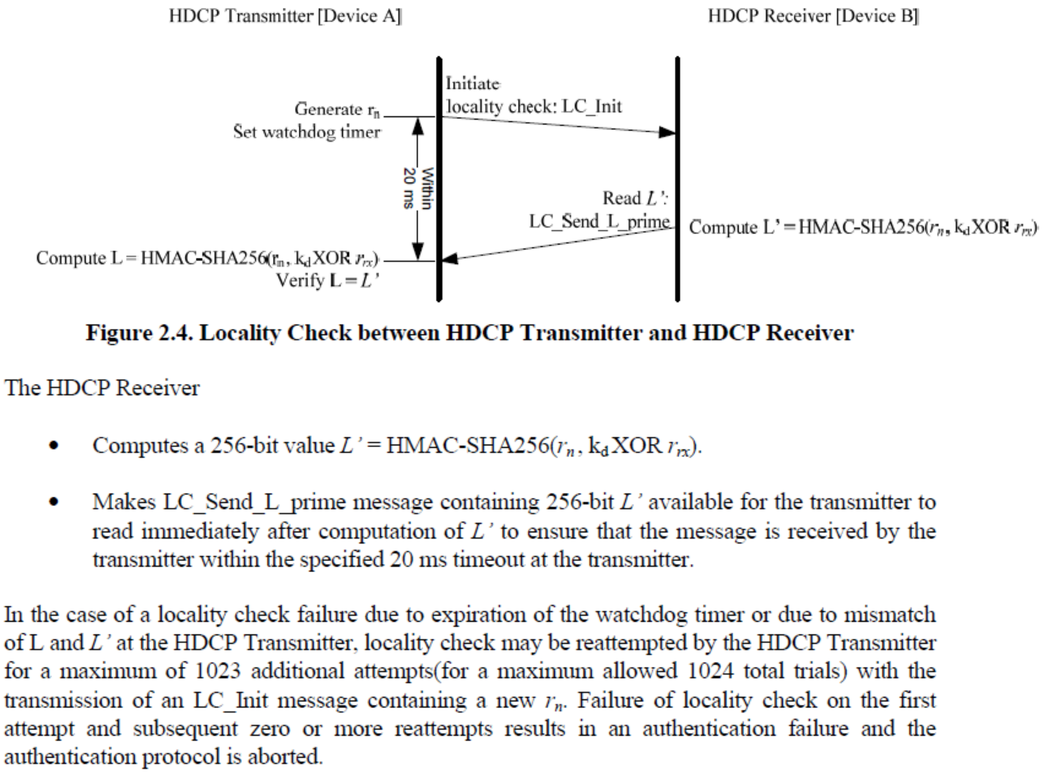


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value L' = HMAC-SHA256(r_n, k_dXOR r_{rx}).
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

See also:

</td>
</tr>
</table>

"determine whether a time difference between providing the first signal and receiving the second signal is less than a predetermined time; and"

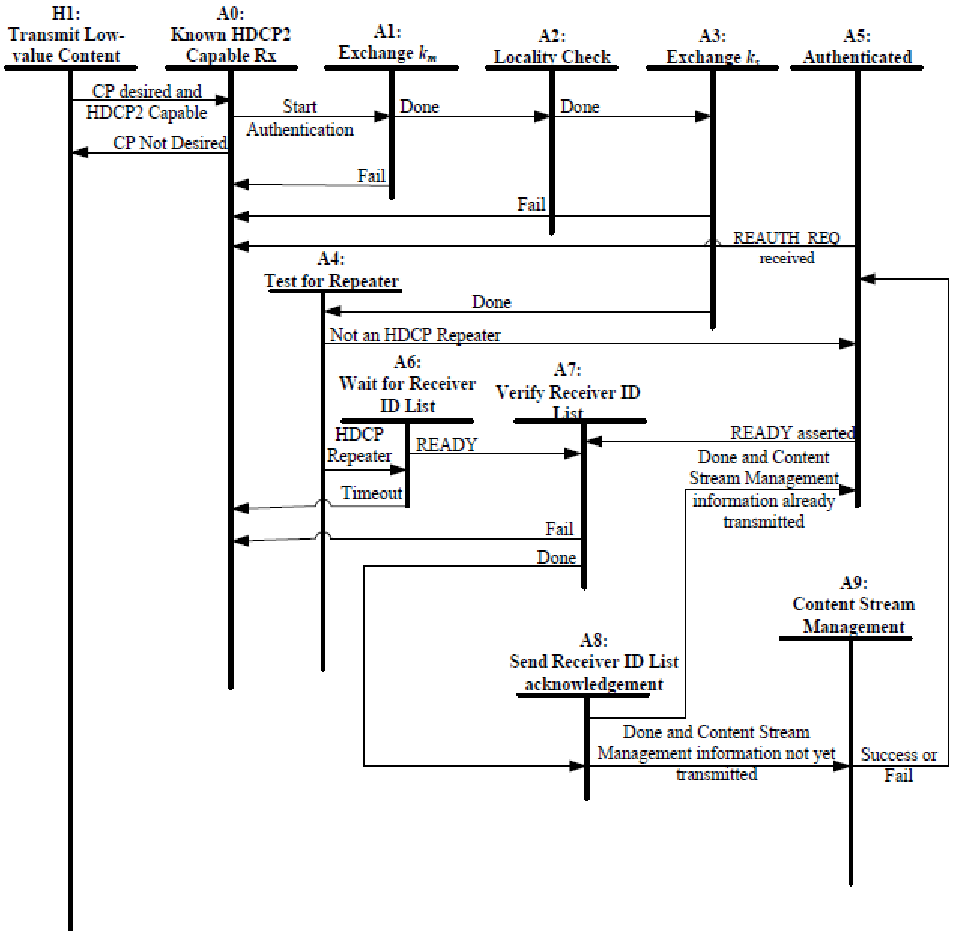


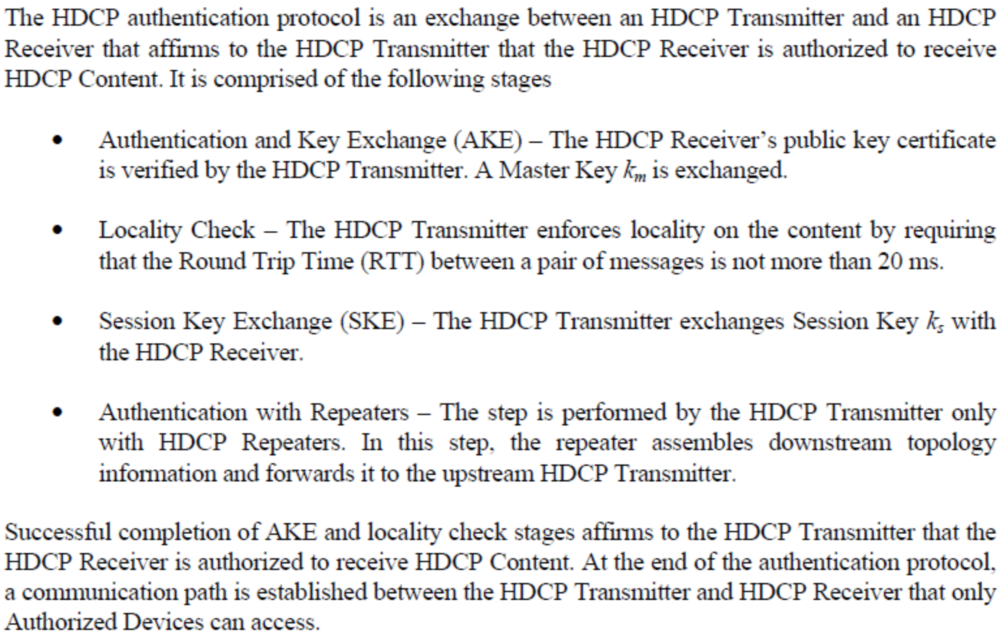
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

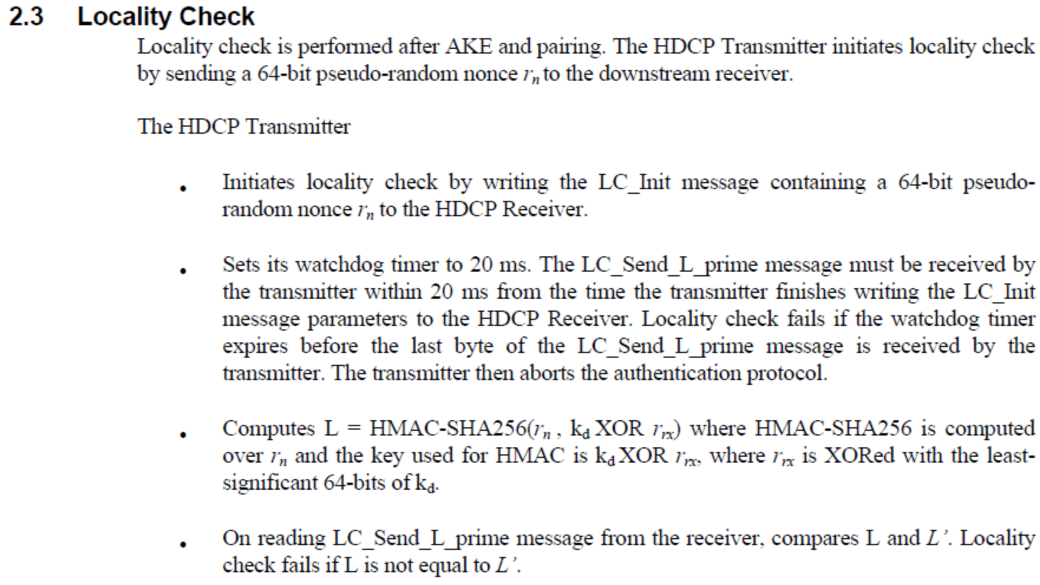
> **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing.
>
> **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

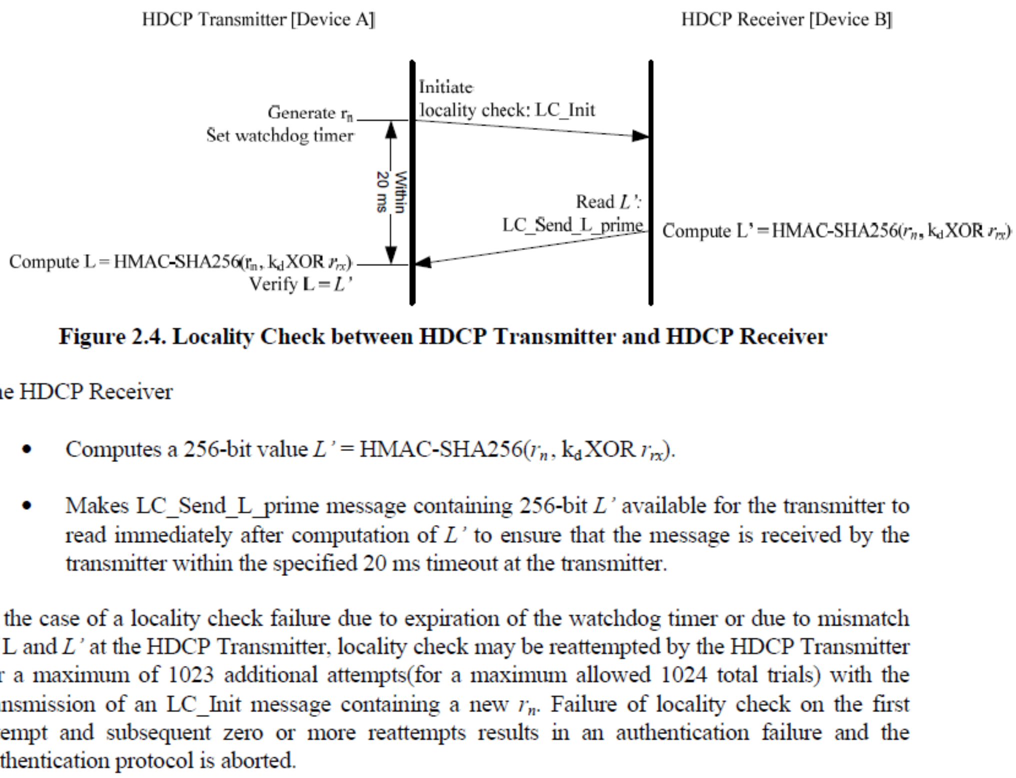
Id. at 28.

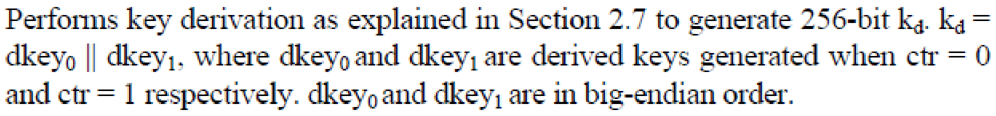
"allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time.	The processor of the Accused Product is arranged to allow the protected content to be provided to the second device at least when the second signal, *e.g.*, L', is determined to be derived from the secret and the time difference is less than the predetermined time. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Accused Product allows the protected content to be provided to the second device at least when, as part of the Locality Check: the L' received via the LC_Send_L_prime message is determined to be derived from the secret (as determined by matching L' to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m)); and the time difference between the providing of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than the predetermined time of 20 ms.

"allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** The HDCP Receiver - Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$. - Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted. *Id*. at 17. The second signal, *e.g.*, L’, is derived from a secret. The value of L’ is derived from k_d.  Compute L’ = HMAC-SHA256(r_n, k_dXOR r_{rx}) *Id*. The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.



Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 14-15.

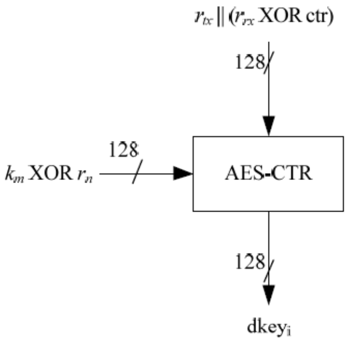
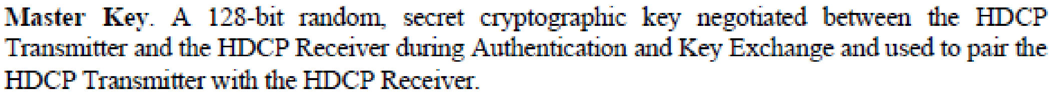


Figure 2.10. Key Derivation

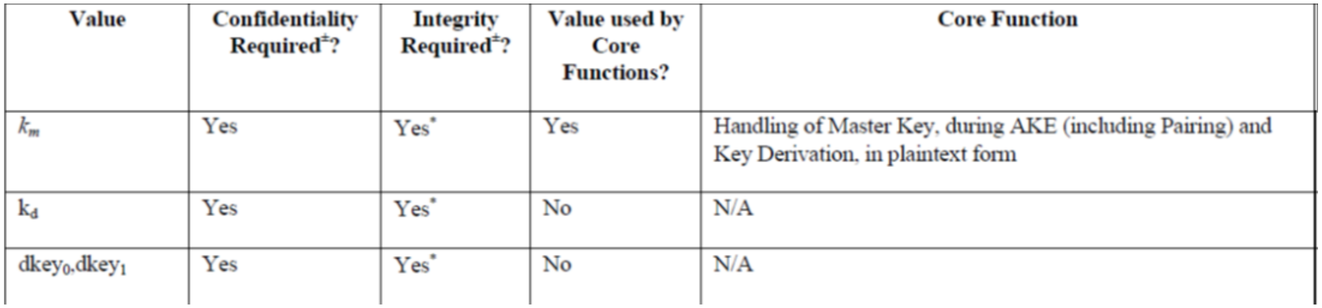
Id. at 25.



Master Key. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver.

Id. at 8.

Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret.

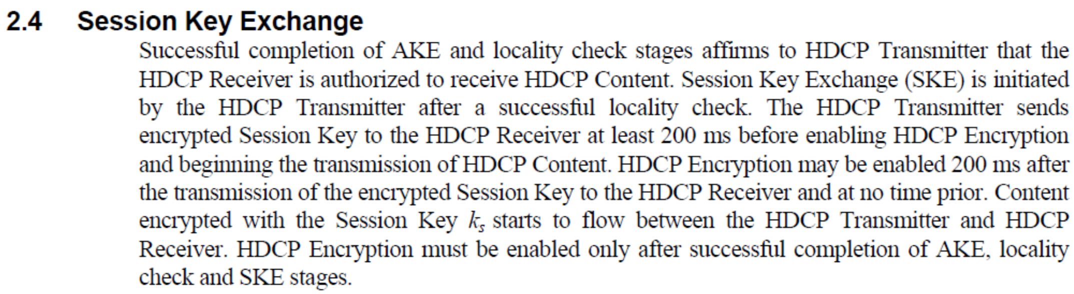


Value	Confidentiality Required$^{\pm}$?	Integrity Required$^{\pm}$?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
$dkey_0$,$dkey_1$	Yes	Yes*	No	N/A

"allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

Id. at 67 (abridged).

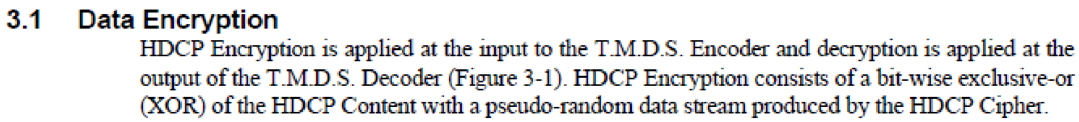
The Accused Product proceeds to session key exchange and providing of the protected content to the second device after successful completion of the AKE stage and Locality Check.



2.4 Session Key Exchange

Successful completion of AKE and locality check stages affirms to HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. Session Key Exchange (SKE) is initiated by the HDCP Transmitter after a successful locality check. The HDCP Transmitter sends encrypted Session Key to the HDCP Receiver at least 200 ms before enabling HDCP Encryption and beginning the transmission of HDCP Content. HDCP Encryption may be enabled 200 ms after the transmission of the encrypted Session Key to the HDCP Receiver and at no time prior. Content encrypted with the Session Key k_s starts to flow between the HDCP Transmitter and HDCP Receiver. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages.

Id. at 17.



3.1 Data Encryption

HDCP Encryption is applied at the input to the T.M.D.S. Encoder and decryption is applied at the output of the T.M.D.S. Decoder (Figure 3-1). HDCP Encryption consists of a bit-wise exclusive-or (XOR) of the HDCP Content with a pseudo-random data stream produced by the HDCP Cipher.

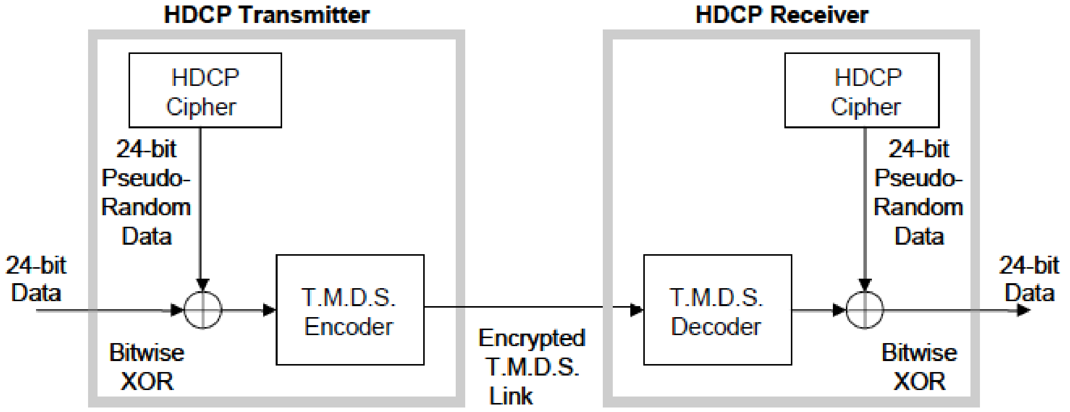


Figure 3-1. HDCP Encryption and Decryption

"allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

Id. at 50.

See also:

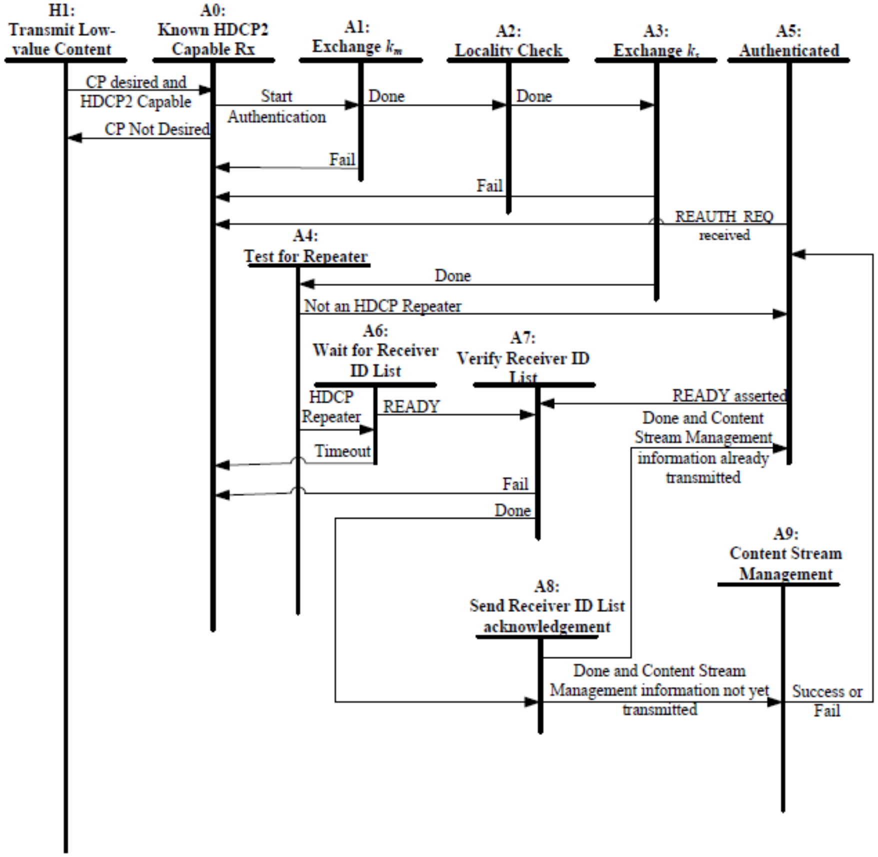


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.

"allow the protected content to be provided to the second device at least when the second signal is determined to be derived from the secret and the time difference is less than the predetermined time."

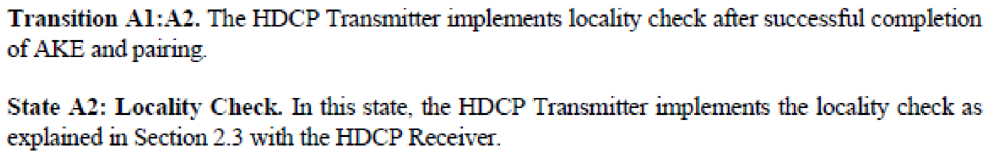
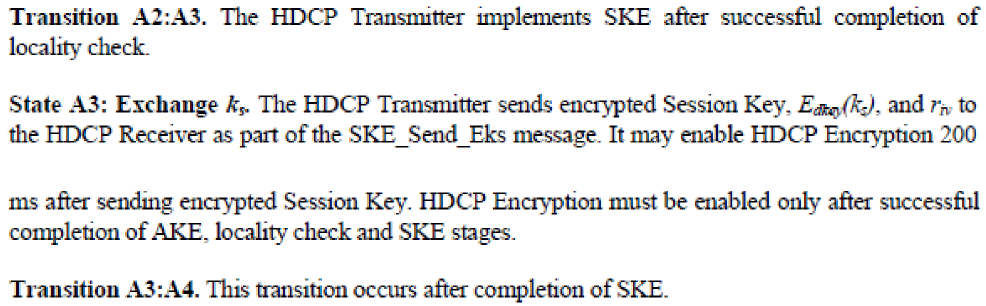
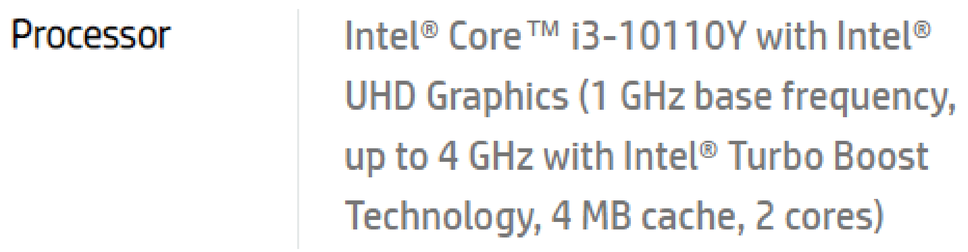
	 **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id*. at 28.  **Transition A2:A3.** The HDCP Transmitter implements SKE after successful completion of locality check. **State A3: Exchange k_s.** The HDCP Transmitter sends encrypted Session Key, $E_{dkey}(k_s)$, and r_{iv} to the HDCP Receiver as part of the SKE_Send_Eks message. It may enable HDCP Encryption 200 ms after sending encrypted Session Key. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages. **Transition A3:A4.** This transition occurs after completion of SKE. *Id*. at 28-29.

EXHIBIT E

U.S. Patent No. 10,091,186

HP Product / Intel Product

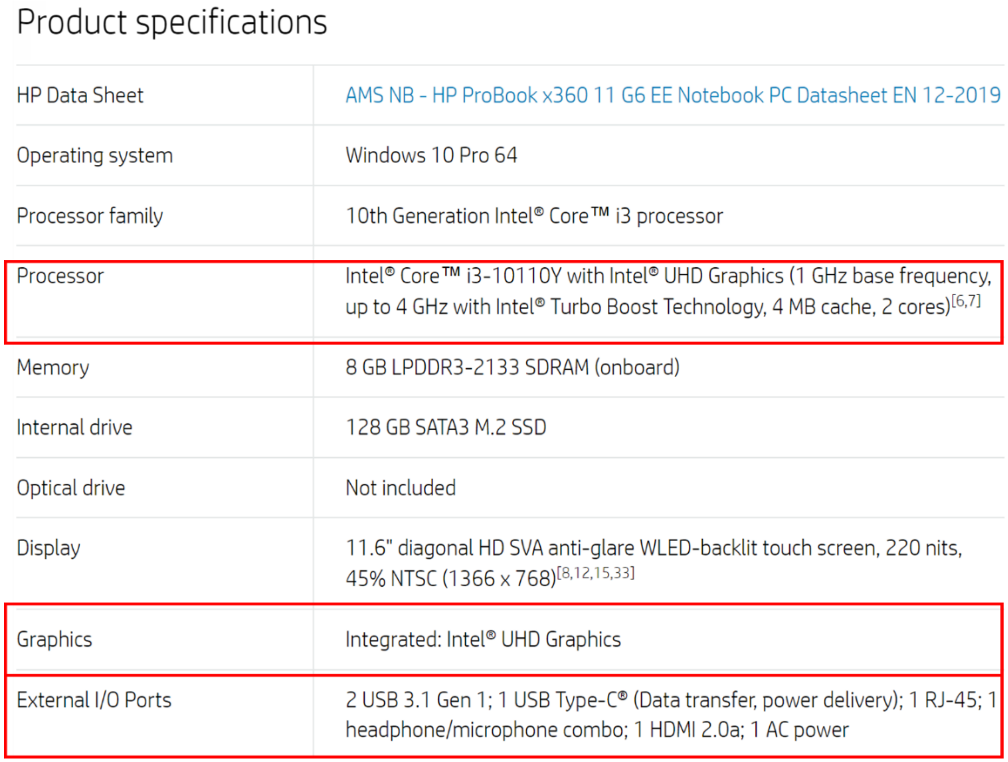
	 **Processor** Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)
HP ProBook x360 11 G6 EE Notebook PC (Product # 3C534UT#ABA) (“HP Product” or “Accused Product”)	Intel video processing system and components thereof including 10th Generation Intel Core i3-10110Y Processor, main board hardware, integrated operating system, middleware, application program, video processing, and/or digital rights management (“DRM”) software that runs on the HP Product (“Intel Product” or “Accused Product”)

U.S. Patent No. 10,091,186: Claim 1

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:	Each of the HP Product and the Intel Product is a first device for controlling delivery of protected content to a second device, and is referred to herein as an “Accused Product.” For example, the HP Product is an HDMI transmitter with HDCP 2.2 for controlling delivery of protected content to another device, such as an HDMI receiver with HDCP 2.2.  HP, HP ProBook x360 11 G6 EE Notebook PC, https://store.hp.com/us/en/pdp/hp-probook-x360-11-g6-ee-notebook-pc. The HP Product includes an HDMI 2.0a port and a 10th Generation Intel® Core™ i3-10110Y Processor (the “Intel Processor”) integrated with the Intel UHD Graphics 615 graphics processor (the “Intel GPU”) that enable delivery of protected content to another device.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"



Product specifications

HP Data Sheet	AMS NB - HP ProBook x360 11 G6 EE Notebook PC Datasheet EN 12-2019
Operating system	Windows 10 Pro 64
Processor family	10th Generation Intel® Core™ i3 processor
Processor	Intel® Core™ i3-10110Y with Intel® UHD Graphics (1 GHz base frequency, up to 4 GHz with Intel® Turbo Boost Technology, 4 MB cache, 2 cores)[6,7]
Memory	8 GB LPDDR3-2133 SDRAM (onboard)
Internal drive	128 GB SATA3 M.2 SSD
Optical drive	Not included
Display	11.6" diagonal HD SVA anti-glare WLED-backlit touch screen, 220 nits, 45% NTSC (1366 x 768)[8,12,15,33]
Graphics	Integrated: Intel® UHD Graphics
External I/O Ports	2 USB 3.1 Gen 1; 1 USB Type-C® (Data transfer, power delivery); 1 RJ-45; 1 headphone/microphone combo; 1 HDMI 2.0a; 1 AC power

Id. *See also* NotebookCheck, Intel Core i3-10110Y, https://www.notebookcheck.net/Intel-Core-i3-10110Y-Laptop-Processor-Comet-Lake-Y.431177.0.html/.

The Intel Processor supports HDCP 2.2 via HDMI 2.0a.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

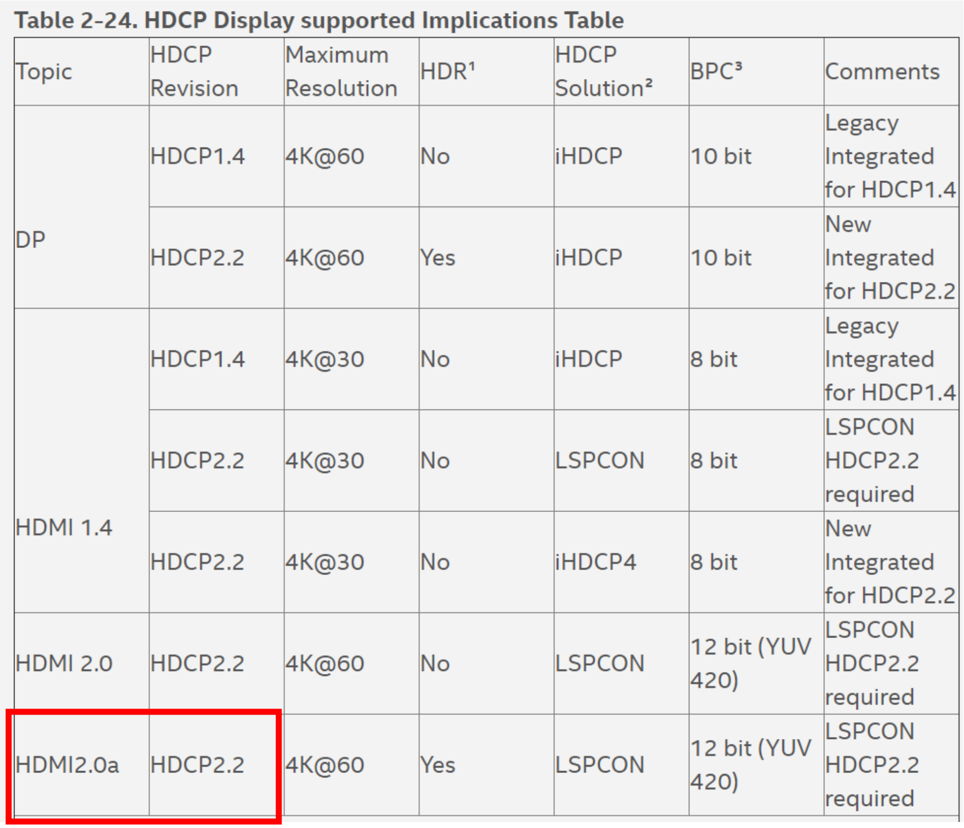
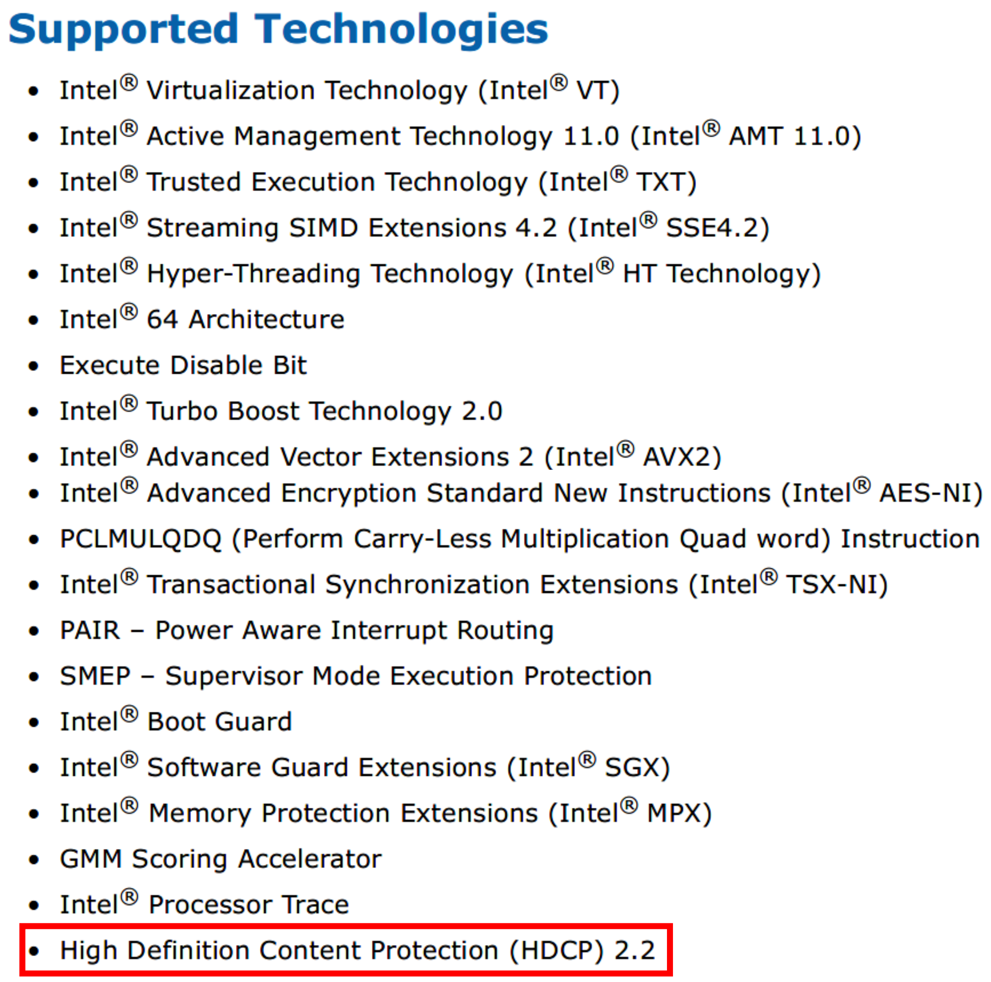


Table 2-24. HDCP Display supported Implications Table

Topic	HDCP Revision	Maximum Resolution	HDR¹	HDCP Solution²	BPC³	Comments
DP	HDCP1.4	4K@60	No	iHDCP	10 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@60	Yes	iHDCP	10 bit	New Integrated for HDCP2.2
HDMI 1.4	HDCP1.4	4K@30	No	iHDCP	8 bit	Legacy Integrated for HDCP1.4
	HDCP2.2	4K@30	No	LSPCON	8 bit	LSPCON HDCP2.2 required
	HDCP2.2	4K@30	No	iHDCP4	8 bit	New Integrated for HDCP2.2
HDMI 2.0	HDCP2.2	4K@60	No	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required
HDMI2.0a	HDCP2.2	4K@60	Yes	LSPCON	12 bit (YUV 420)	LSPCON HDCP2.2 required

Intel, How to enable High Dynamic Range?, https://www.intel.com/content/www/us/en/support/articles/000032112/graphics/graphics-for-7th-generation-intel-processors.html.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"



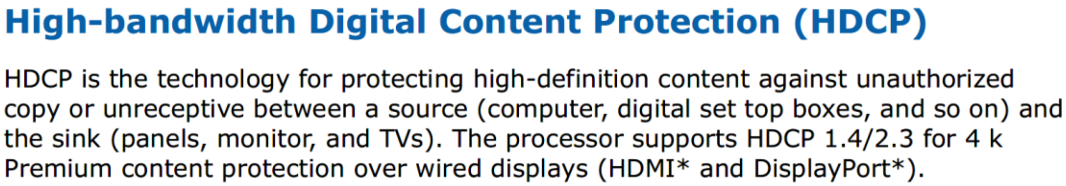
Supported Technologies

- Intel® Virtualization Technology (Intel® VT)
- Intel® Active Management Technology 11.0 (Intel® AMT 11.0)
- Intel® Trusted Execution Technology (Intel® TXT)
- Intel® Streaming SIMD Extensions 4.2 (Intel® SSE4.2)
- Intel® Hyper-Threading Technology (Intel® HT Technology)
- Intel® 64 Architecture
- Execute Disable Bit
- Intel® Turbo Boost Technology 2.0
- Intel® Advanced Vector Extensions 2 (Intel® AVX2)
- Intel® Advanced Encryption Standard New Instructions (Intel® AES-NI)
- PCLMULQDQ (Perform Carry-Less Multiplication Quad word) Instruction
- Intel® Transactional Synchronization Extensions (Intel® TSX-NI)
- PAIR – Power Aware Interrupt Routing
- SMEP – Supervisor Mode Execution Protection
- Intel® Boot Guard
- Intel® Software Guard Extensions (Intel® SGX)
- Intel® Memory Protection Extensions (Intel® MPX)
- GMM Scoring Accelerator
- Intel® Processor Trace
- High Definition Content Protection (HDCP) 2.2

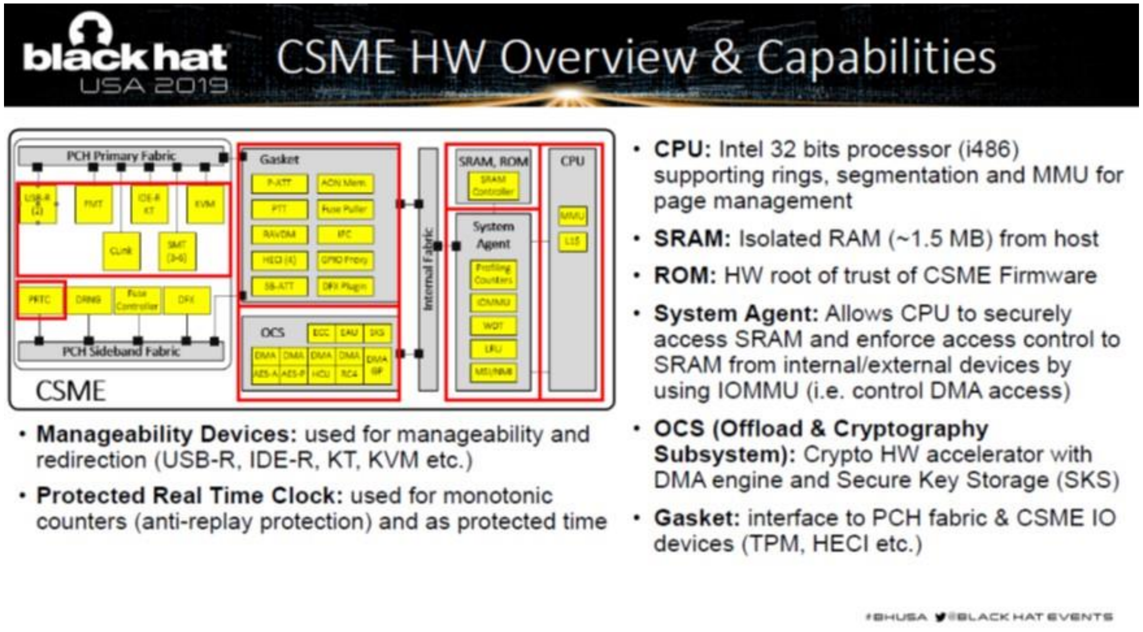
Intel, 10th Generation Intel Core Processors, Datasheet, Volume 1 or 2 (Jul. 2020, rev. 5), *available at* https://cdrdv2.intel.com/v1/dl/getContent/615211, at 11-12.

"HDCP is the technology for protecting high-definition content against unauthorized copy … between a source … and the sink …. The [Intel] [P]rocessor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*)."

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

	 **High-bandwidth Digital Content Protection (HDCP)** HDCP is the technology for protecting high-definition content against unauthorized copy or unreceptive between a source (computer, digital set top boxes, and so on) and the sink (panels, monitor, and TVs). The processor supports HDCP 1.4/2.3 for 4 k Premium content protection over wired displays (HDMI* and DisplayPort*). *Id.* at 44 Intel's "UHD" processor nomenclature also indicates support for HDCP 2.2: Another change from 7 Gen to 8 Gen will be in the graphics. Intel is upgrading the nomenclature of the integrated graphics from HD 620 to UHD 620, indicating that the silicon is suited for 4K playback and processing. During our pre-briefing it was categorically stated several times that there was no change between the two, however we have since confirmed that the new chips will come with HDCP 2.2 support as standard for DP1.2a, removing the need for an external LSPCON for this feature. Other than this display controller change however, it appears that these new UHD iGPUs are architecturally the same as their HD predecessors. https://www.anandtech.com/show/11738/intel-launches-8th-generation-cpus-starting-with-kaby-lake-refresh-for-15w-mobile. HDCP 2.2 is implemented in Intel-based systems with Core-i series Processors within the Converged Security & Manageability Engine (CSME) also known as the Management Engine (ME). The CSME contains a processor (x86 core) which executes instructions including but not limited to the uKernel/OS, drivers, services, and applications for the CSME.

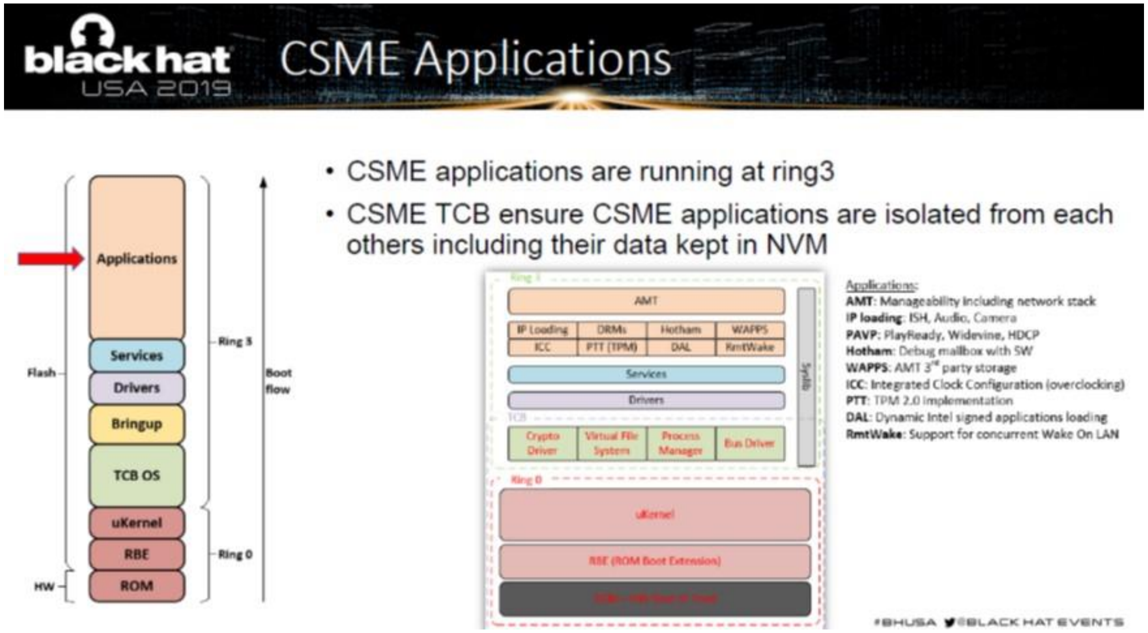
"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"



CSME HW Overview & Capabilities

- **CPU:** Intel 32 bits processor (i486) supporting rings, segmentation and MMU for page management
- **SRAM:** Isolated RAM (~1.5 MB) from host
- **ROM:** HW root of trust of CSME Firmware
- **System Agent:** Allows CPU to securely access SRAM and enforce access control to SRAM from internal/external devices by using IOMMU (i.e. control DMA access)
- **OCS (Offload & Cryptography Subsystem):** Crypto HW accelerator with DMA engine and Secure Key Storage (SKS)
- **Gasket:** interface to PCH fabric & CSME IO devices (TPM, HECI etc.)
- **Manageability Devices:** used for manageability and redirection (USB-R, IDE-R, KT, KVM etc.)
- **Protected Real Time Clock:** used for monotonic counters (anti-replay protection) and as protected time

Behind the Scenes of Intel Security and Manageability Engine, blackhat USA 2019 ("CSME") at 7.



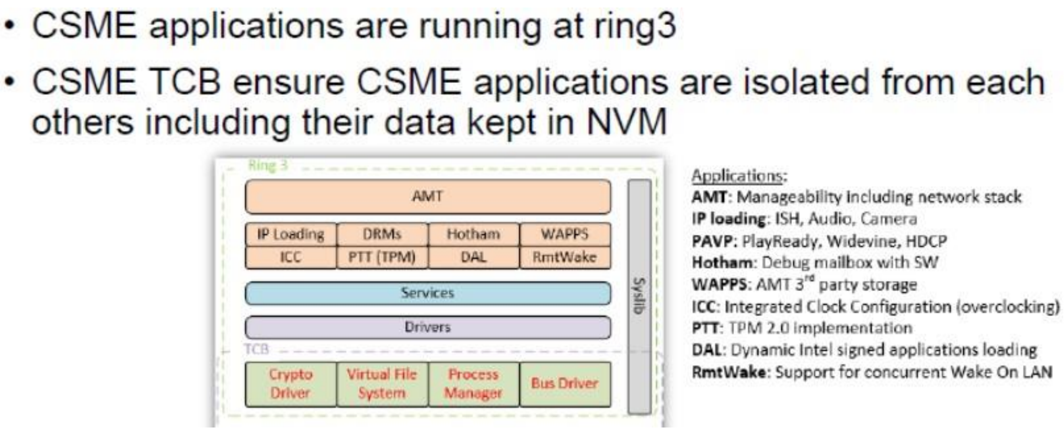
CSME Applications

- CSME applications are running at ring3
- CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM

Applications:
AMT: Manageability including network stack
IP loading: ISH, Audio, Camera
PAVP: PlayReady, Widevine, HDCP
Hotham: Debug mailbox with SW
WAPPS: AMT 3rd party storage
ICC: Integrated Clock Configuration (overclocking)
PTT: TPM 2.0 implementation
DAL: Dynamic Intel signed applications loading
RmtWake: Support for concurrent Wake On LAN

Id. at 23.

One such application is "PAVP" which provides HDCP capabilities within the Intel processor.

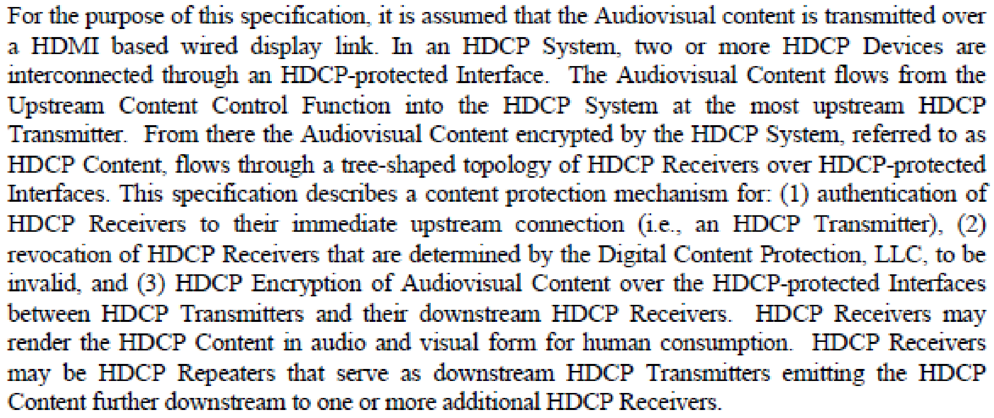
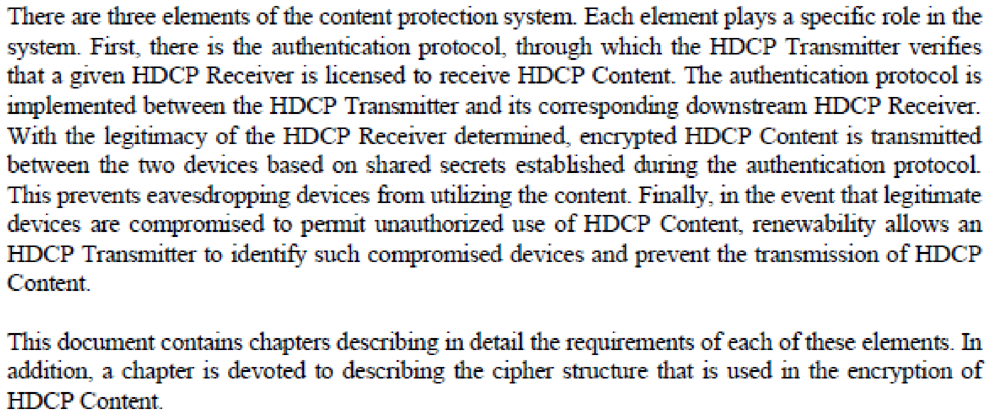


- CSME applications are running at ring3
- CSME TCB ensure CSME applications are isolated from each others including their data kept in NVM

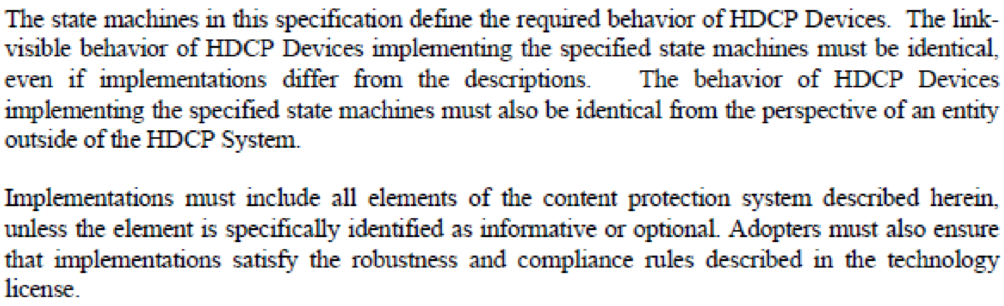
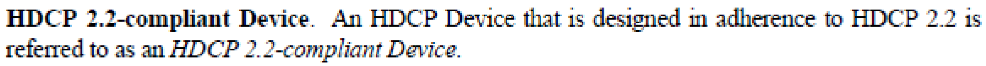
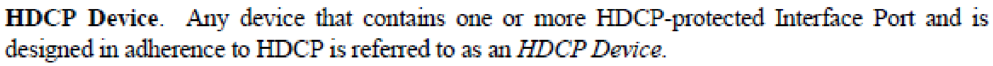
Applications:
AMT: Manageability including network stack
IP loading: ISH, Audio, Camera
PAVP: PlayReady, Widevine, HDCP
Hotham: Debug mailbox with SW
WAPPS: AMT 3rd party storage
ICC: Integrated Clock Configuration (overclocking)
PTT: TPM 2.0 implementation
DAL: Dynamic Intel signed applications loading
RmtWake: Support for concurrent Wake On LAN

Id.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

	Upon information and belief, the Accused Product is compliant with the High-bandwidth Digital Content Protection System Revision 2.2 ("HDCP 2.2") protocol. The Accused Product supports HDCP 2.2 for protecting content between devices.  For the purpose of this specification, it is assumed that the Audiovisual content is transmitted over a HDMI based wired display link. In an HDCP System, two or more HDCP Devices are interconnected through an HDCP-protected Interface. The Audiovisual Content flows from the Upstream Content Control Function into the HDCP System at the most upstream HDCP Transmitter. From there the Audiovisual Content encrypted by the HDCP System, referred to as HDCP Content, flows through a tree-shaped topology of HDCP Receivers over HDCP-protected Interfaces. This specification describes a content protection mechanism for: (1) authentication of HDCP Receivers to their immediate upstream connection (i.e., an HDCP Transmitter), (2) revocation of HDCP Receivers that are determined by the Digital Content Protection, LLC, to be invalid, and (3) HDCP Encryption of Audiovisual Content over the HDCP-protected Interfaces between HDCP Transmitters and their downstream HDCP Receivers. HDCP Receivers may render the HDCP Content in audio and visual form for human consumption. HDCP Receivers may be HDCP Repeaters that serve as downstream HDCP Transmitters emitting the HDCP Content further downstream to one or more additional HDCP Receivers. High-bandwidth Digital Content Protection System Mapping HDCP to HDMI Revision 2.2 13 February, 2013 ("HDMI HDCP 2.2") at 5.  There are three elements of the content protection system. Each element plays a specific role in the system. First, there is the authentication protocol, through which the HDCP Transmitter verifies that a given HDCP Receiver is licensed to receive HDCP Content. The authentication protocol is implemented between the HDCP Transmitter and its corresponding downstream HDCP Receiver. With the legitimacy of the HDCP Receiver determined, encrypted HDCP Content is transmitted between the two devices based on shared secrets established during the authentication protocol. This prevents eavesdropping devices from utilizing the content. Finally, in the event that legitimate devices are compromised to permit unauthorized use of HDCP Content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of HDCP Content. This document contains chapters describing in detail the requirements of each of these elements. In addition, a chapter is devoted to describing the cipher structure that is used in the encryption of HDCP Content. *Id*. at 9.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

	The Accused Product is an HDCP Device, and more specifically an HDCP 2.2-compliant Device, capable of functioning as an HDCP Transmitter and that implements required functionality of HDMI HDCP 2.2 including the functions required by the HDCP Transmitter State Diagram.  The state machines in this specification define the required behavior of HDCP Devices. The link-visible behavior of HDCP Devices implementing the specified state machines must be identical, even if implementations differ from the descriptions. The behavior of HDCP Devices implementing the specified state machines must also be identical from the perspective of an entity outside of the HDCP System. Implementations must include all elements of the content protection system described herein, unless the element is specifically identified as informative or optional. Adopters must also ensure that implementations satisfy the robustness and compliance rules described in the technology license. *Id*. at 5.  **HDCP 2.2-compliant Device**. An HDCP Device that is designed in adherence to HDCP 2.2 is referred to as an *HDCP 2.2-compliant Device*. *Id*. at 6.  **HDCP Device**. Any device that contains one or more HDCP-protected Interface Port and is designed in adherence to HDCP is referred to as an *HDCP Device*. *Id*. at 7.

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

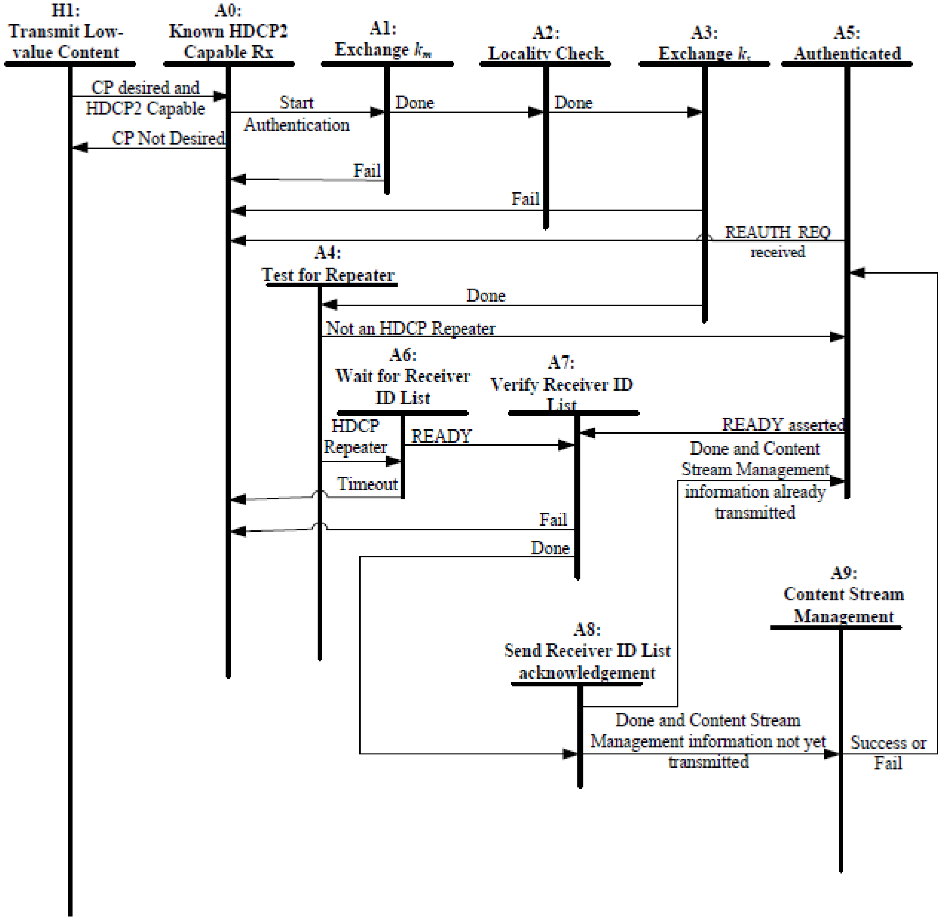
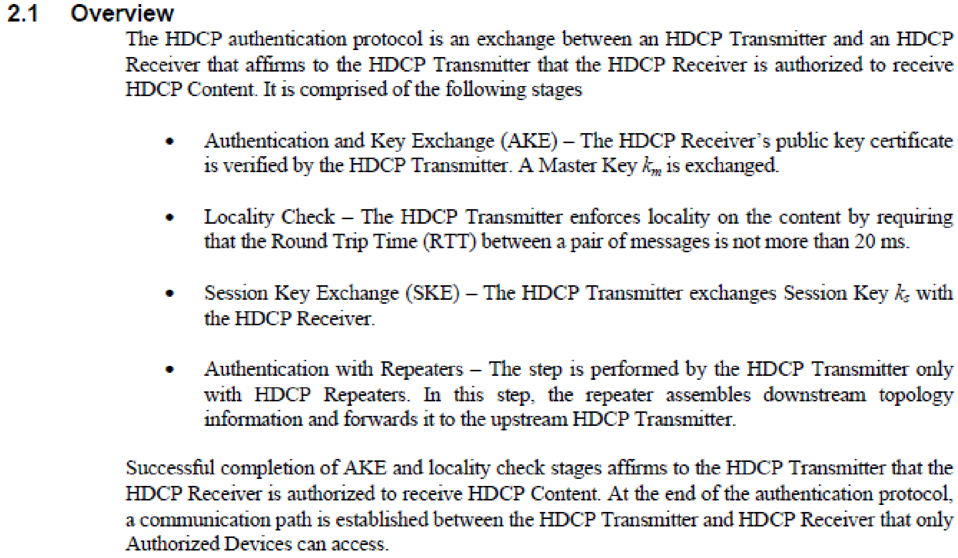
<table>
<tr>
<td></td>
<td>


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27-30.

The Accused Product controls delivery of protected content to a second device.
</td>
</tr>
</table>

"1. A first device for controlling delivery of protected content to a second device, the first device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

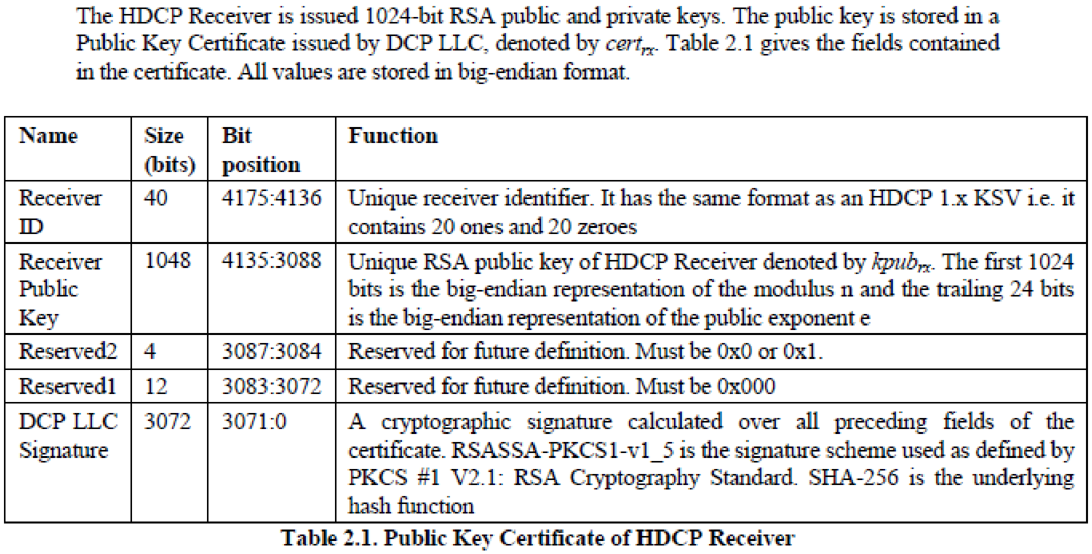
	 **2.1 Overview** The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. *Id*. at 11.

"receive a second device certificate from the second device prior to sending a first signal;"

receive a second device certificate from the second device prior to sending a first signal;

The instructions of the Accused Product are arranged to receive a second device certificate, *e.g.*, $cert_{rx}$, from the second device (receiver) as part of the Authentication and Key Exchange (AKE) stage of the HDCP 2.2 protocol and prior to sending a first signal, *e.g.*, the LC_Init message including r_n.

The certificate, $cert_{rx}$, includes a Receiver ID for the second device, Receiver Public Key for the second device, and a cryptographic signature, amongst other information.

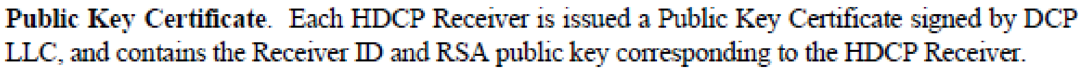


The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

Name	**Size (bits)**	**Bit position**	**Function**
Receiver ID	40	4175:4136	Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes
Receiver Public Key	1048	4135:3088	Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e
Reserved2	4	3087:3084	Reserved for future definition. Must be 0x0 or 0x1.
Reserved1	12	3083:3072	Reserved for future definition. Must be 0x000
DCP LLC Signature	3072	3071:0	A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function

Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.



Public Key Certificate. Each HDCP Receiver is issued a Public Key Certificate signed by DCP LLC, and contains the Receiver ID and RSA public key corresponding to the HDCP Receiver.

Id. at 8.

The Accused Product receives the certificate from the second device as part of the AKE stage, irrespective of whether the Accused Product has a Master Key k_m stored corresponding to the Receiver ID.

"receive a second device certificate from the second device prior to sending a first signal;"

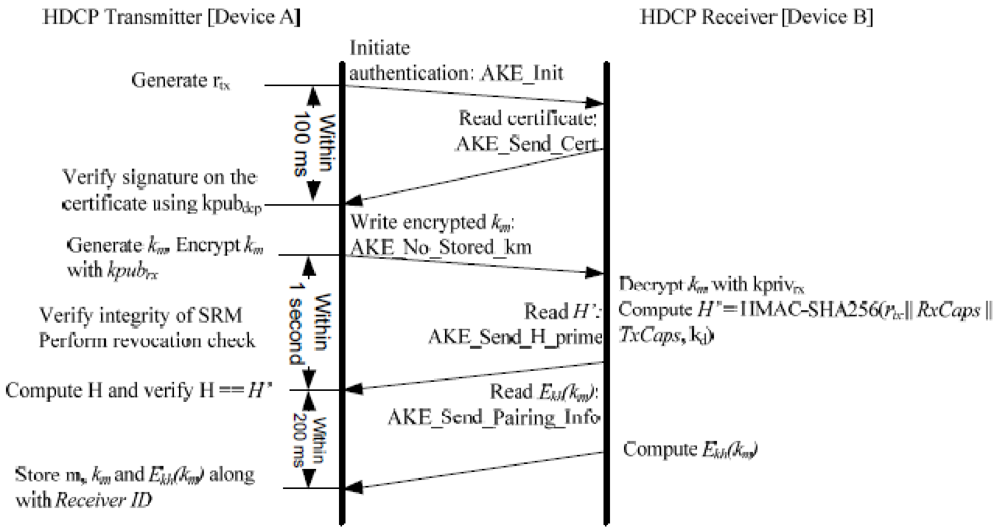


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

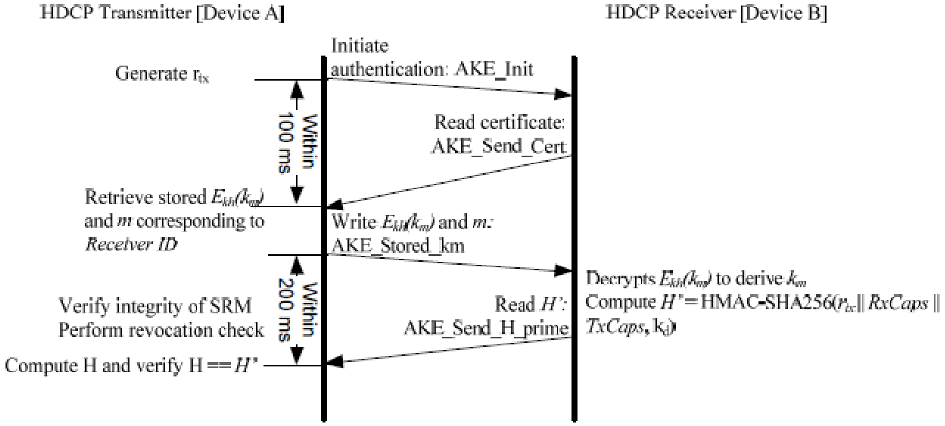
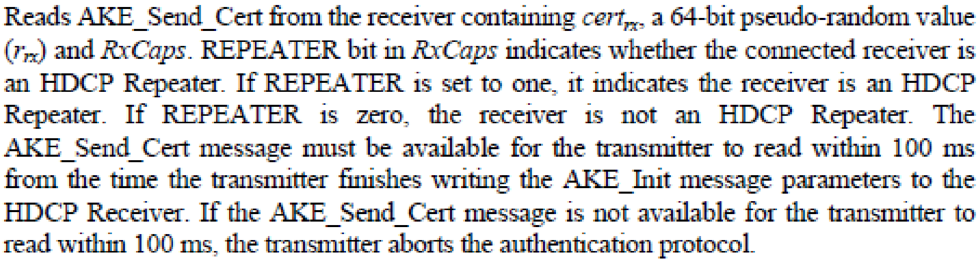


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

Id.

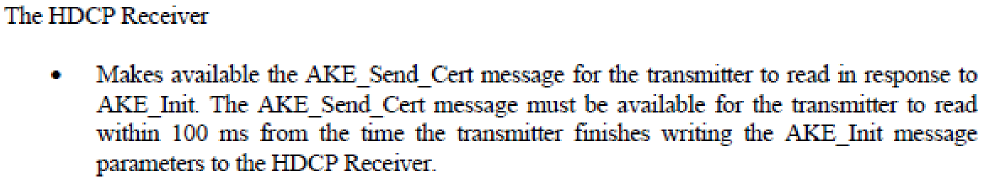
"receive a second device certificate from the second device prior to sending a first signal;"

The Accused Product receives the certificate from the second device as part of the AKE_Send_Cert message.



Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol.

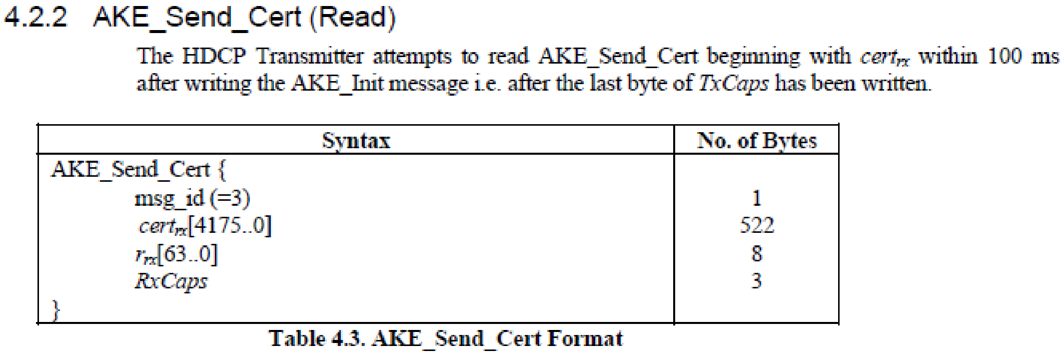
Id. at 13.



The HDCP Receiver

- Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

Id. at 14.



4.2.2 AKE_Send_Cert (Read)

The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

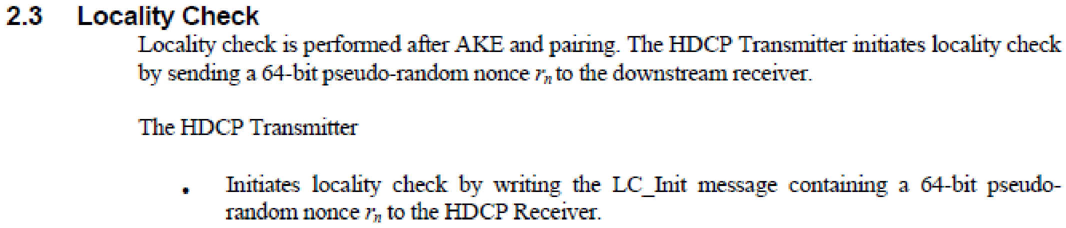
Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

The Accused Product receives the certificate from the second device during the AKE stage prior to sending a first signal, *e.g.*, the LC_Init message including r_n, as part of a Locality Check.

"receive a second device certificate from the second device prior to sending a first signal;"

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter - Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. *Id.* at 16. *See also*:

"receive a second device certificate from the second device prior to sending a first signal;"

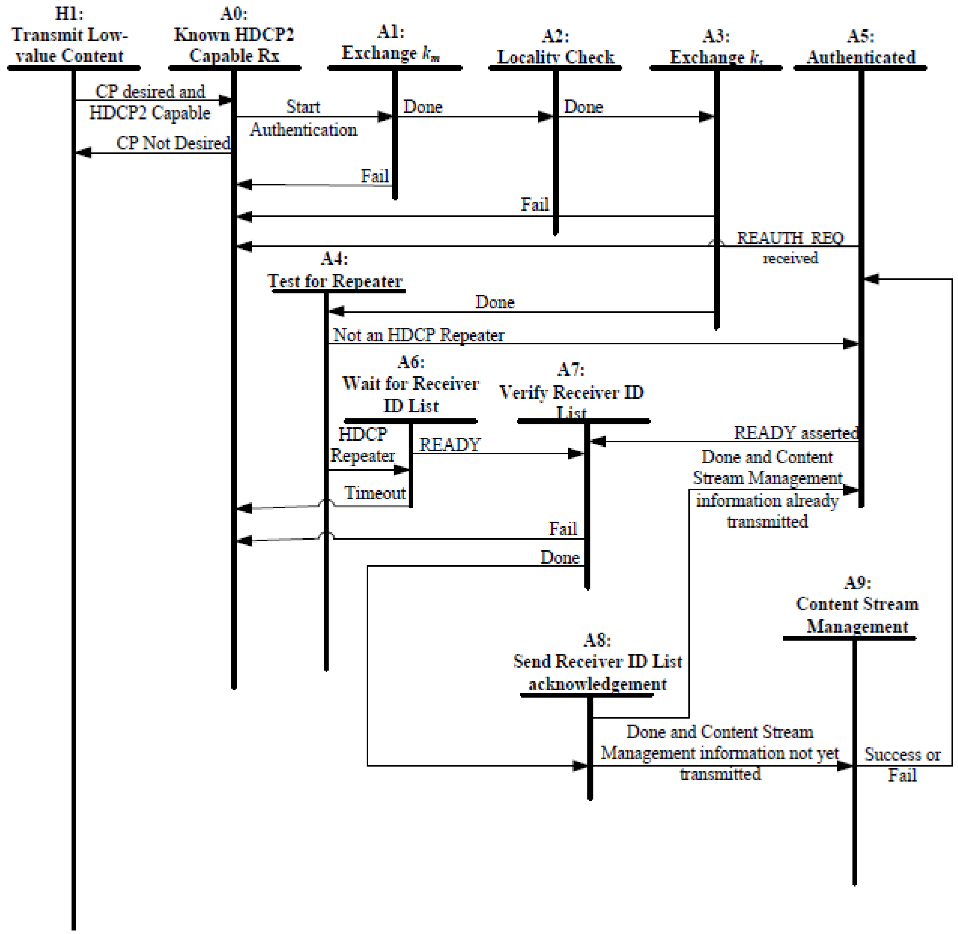
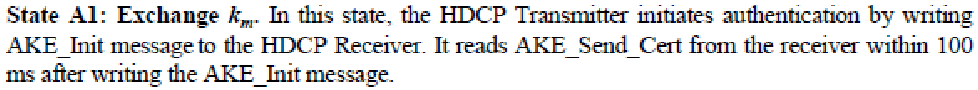


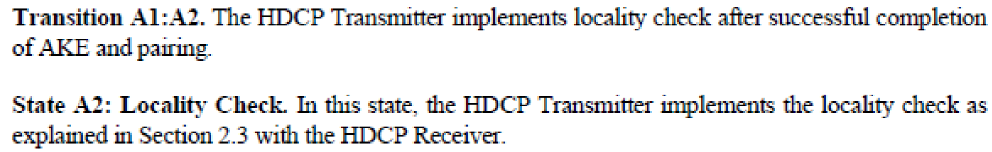
Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

Id. at 27.



State A1: Exchange k_m. In this state, the HDCP Transmitter initiates authentication by writing AKE_Init message to the HDCP Receiver. It reads AKE_Send_Cert from the receiver within 100 ms after writing the AKE_Init message.

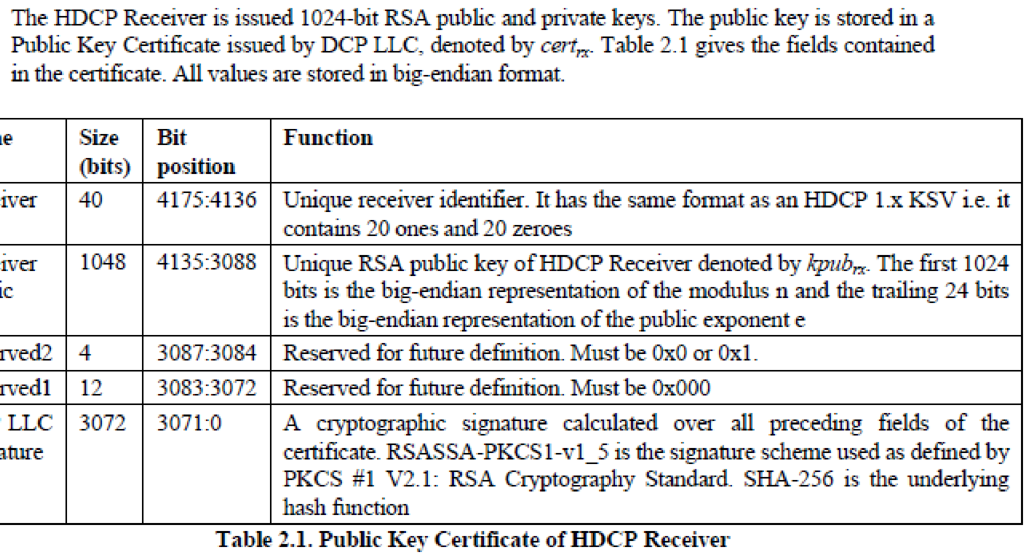
Id. at 28.

	 **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id.*

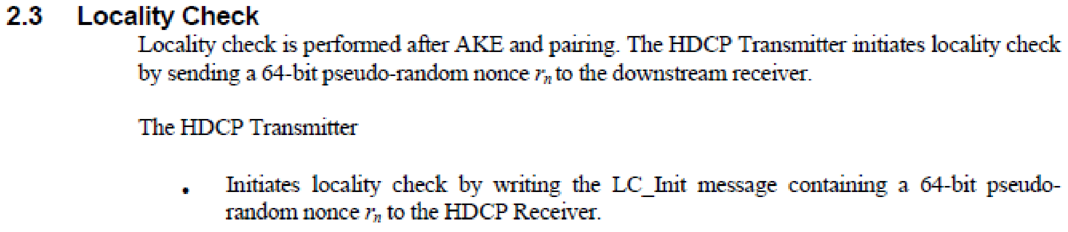
"provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;"

<table>
<tr>
<td>provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;</td>
<td>
The instructions of the Accused Product are arranged to provide the first signal *e.g.*, the LC_Init message including r_n, to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule.

The Accused Product provides the LC_Init message including r_n when the Accused Product determines in the Authentication and Key Exchange (AKE) stage that the certificate, $cert_{rx}$, indicates that the second device is compliant with at least one compliance rule. For example, the certificate, $cert_{rx}$, includes a Receiver ID, Receiver Public Key, and a cryptographic signature.


The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.

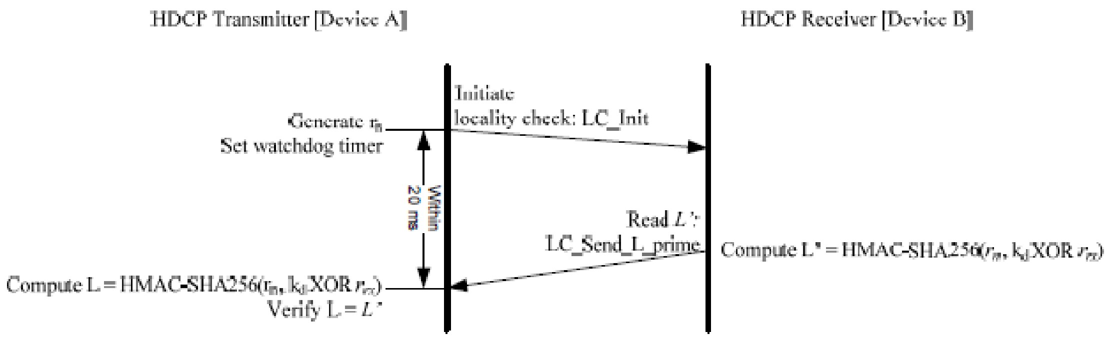

2.3 Locality Check
Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

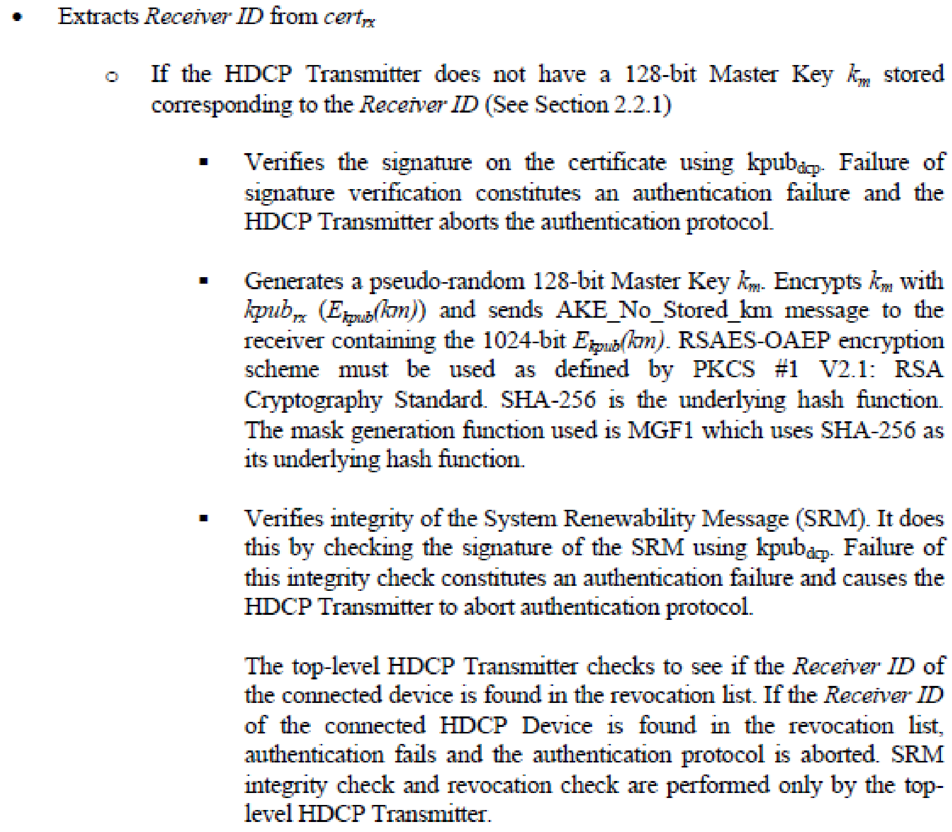
• Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.

Id. at 16.
</td>
</tr>
</table>

"provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;"

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** *Id*. at 17. The Accused Product provides the LC_Init message to the second device when the certificate indicates, for example, that the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.

"provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;"



- Extracts *Receiver ID* from *cert*$_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using kpub$_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using kpub$_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

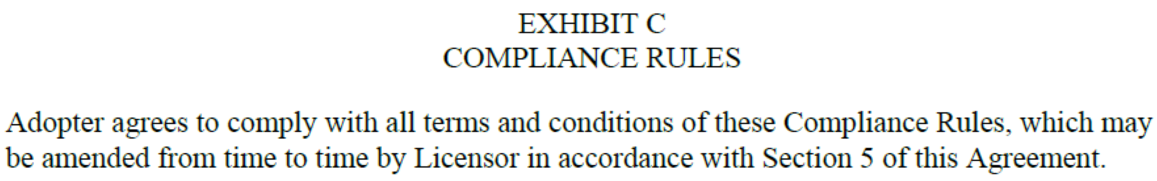
Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with compliance rules of the HDCP specification.

1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

HDCP License Agreement, March 6, 2017, at 2, *available at* https://digital-cp.com/sites/default/files/HDCP%20License%20Agreement_March%206%2C%202017_FOR%20REVIEW%20ONLY.pdf.

"provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;"

	 EXHIBIT C COMPLIANCE RULES Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement. *Id*. at Exhibit C. *See also*:

U.S. Patent No. 10,091,186: Claim 1

"provide the first signal to the second device when the second device certificate indicates that the second device is compliant with at least one compliance rule;"

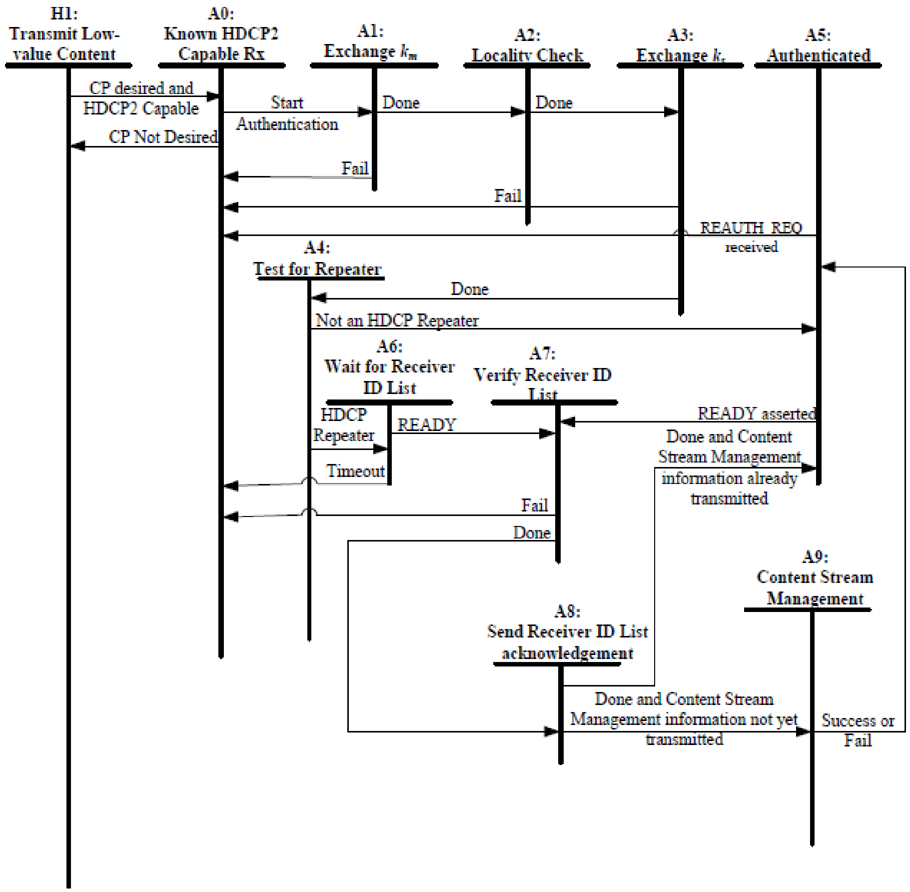
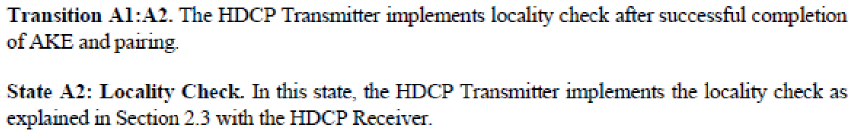


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

HDMI HDCP 2.2 at 27.

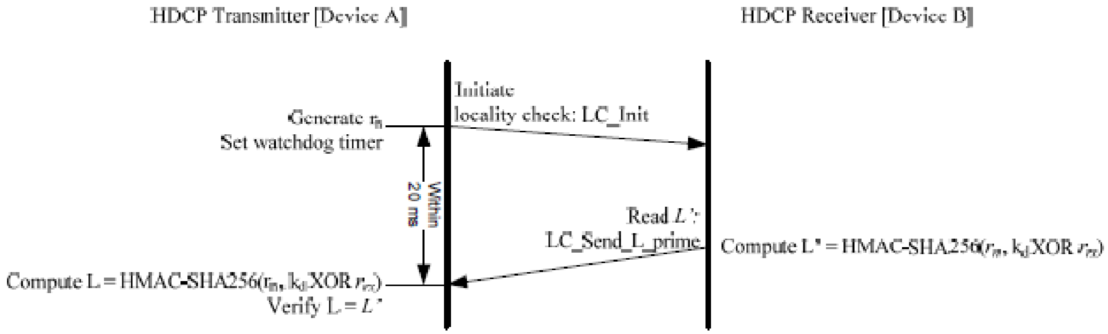
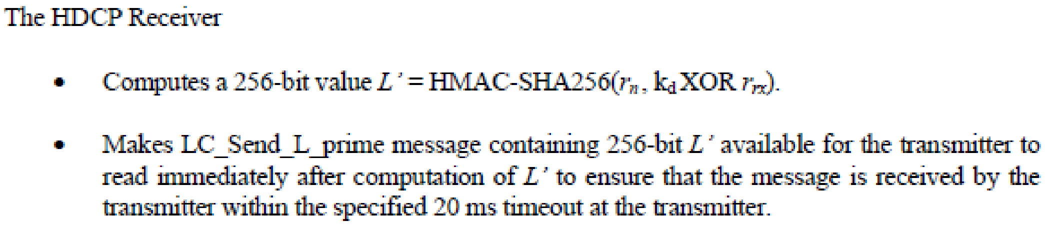


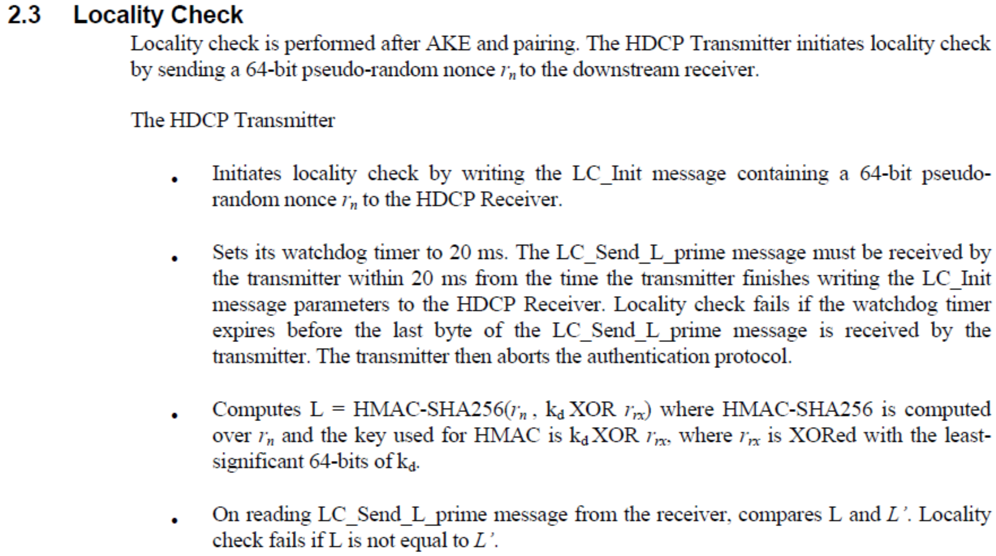
Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

Id. at 28.

"receive a second signal from the second device after providing the first signal; and"

receive a second signal from the second device after providing the first signal; and	The instructions of the Accused Product are arranged to receive a second signal, *e.g.*, the LC_Send_L_prime message including L', from the second device after providing the first signal, *e.g.*, the LC_Init message including r_n.  **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** HDMI HDCP 2.2 at 17.  The HDCP Receiver • Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$. • Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter. *Id.*



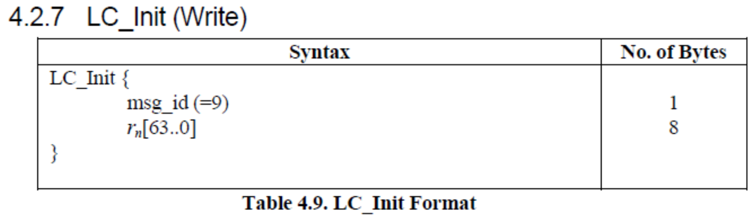
2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and *L'*. Locality check fails if L is not equal to *L'*.

Id. at 16.

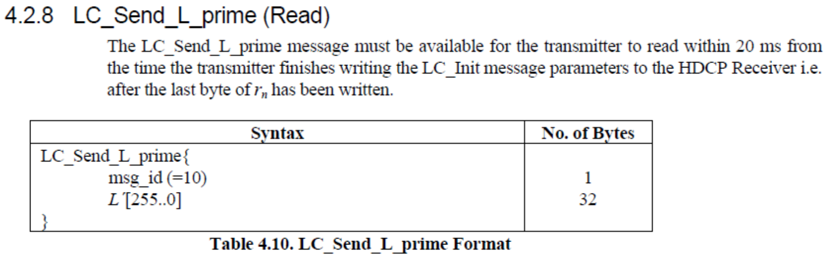


4.2.7 LC_Init (Write)

Syntax	No. of Bytes
LC_Init {	
msg_id (=9)	1
r_n[63..0]	8
}	

Table 4.9. LC_Init Format

Id. at 59.



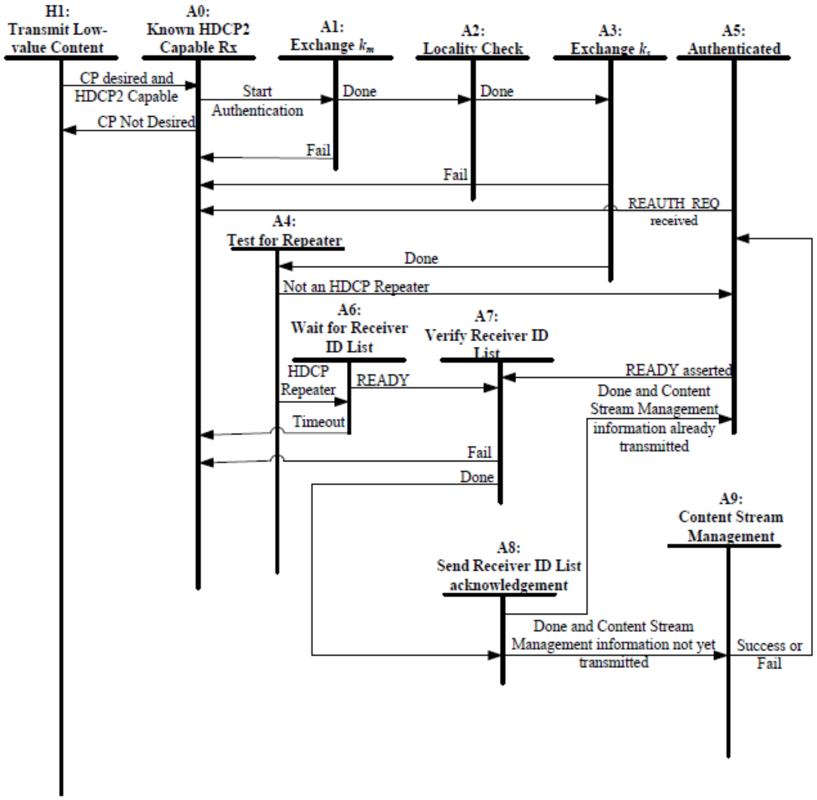
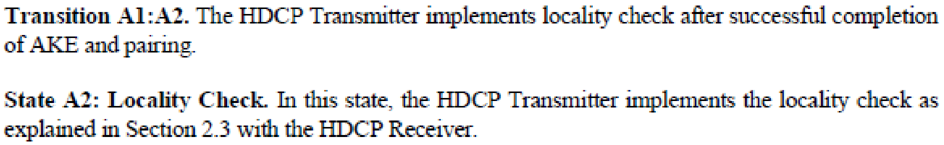
4.2.8 LC_Send_L_prime (Read)

The LC_Send_L_prime message must be available for the transmitter to read within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver i.e. after the last byte of r_n has been written.

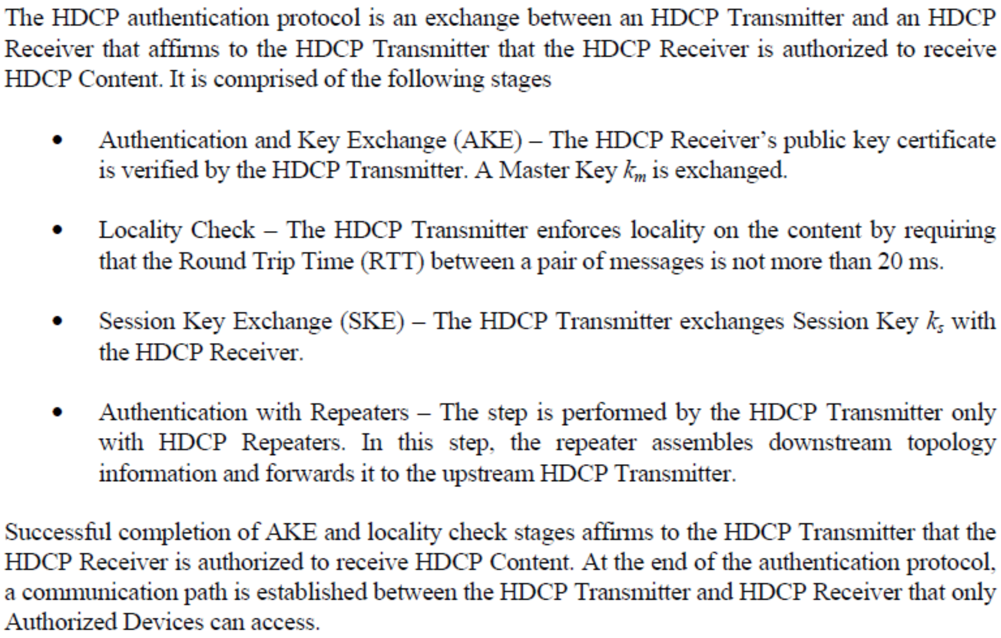
Syntax	No. of Bytes
LC_Send_L_prime{	
msg_id (=10)	1
L'[255..0]	32
}	

Table 4.10. LC_Send_L_prime Format

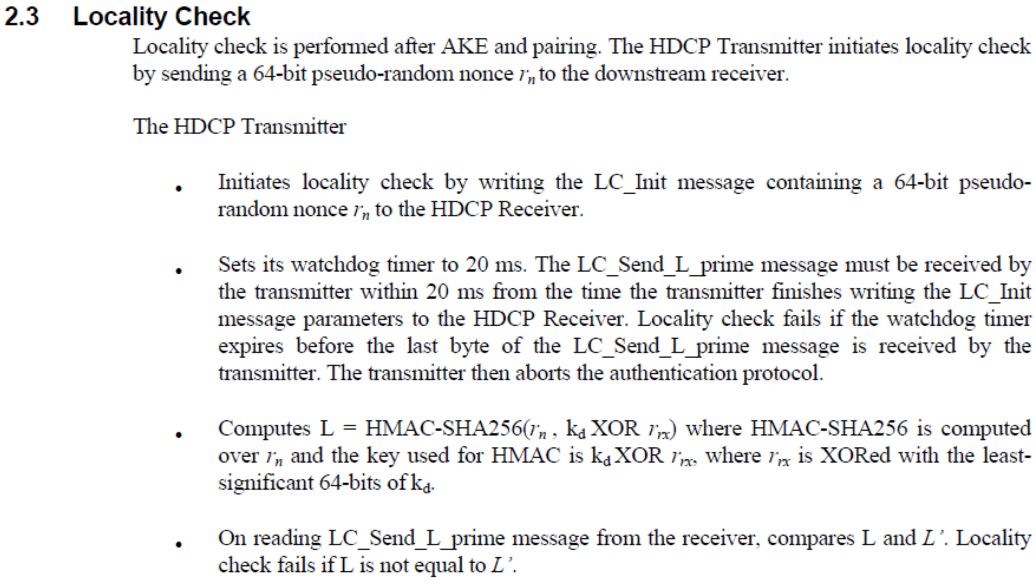
Id.

	See also:  **Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram** *Id*. at 27.  **Transition A1:A2.** The HDCP Transmitter implements locality check after successful completion of AKE and pairing. **State A2: Locality Check.** In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver. *Id*. at 28.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,	The instructions of the Accused Product are arranged to provide the protected content to the second device when the second signal, *e.g.*, L’, is derived from a secret and a time between the providing of the first signal, *e.g.*, the LC_Init message including r_n, and the receiving of the second signal is less than a predetermined time. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for the Accused Product to provide protected content to the second device.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver’s public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Accused Product provides protected content to the second device when, as part of the Locality Check: the L’ received via the LC_Send_L_prime message is derived from a secret (as determined by matching L’ to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m)); and a time between the providing of the LC_Init message and receiving L’ via the LC_Send_L_prime message is less than a predetermined time of 20 ms.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

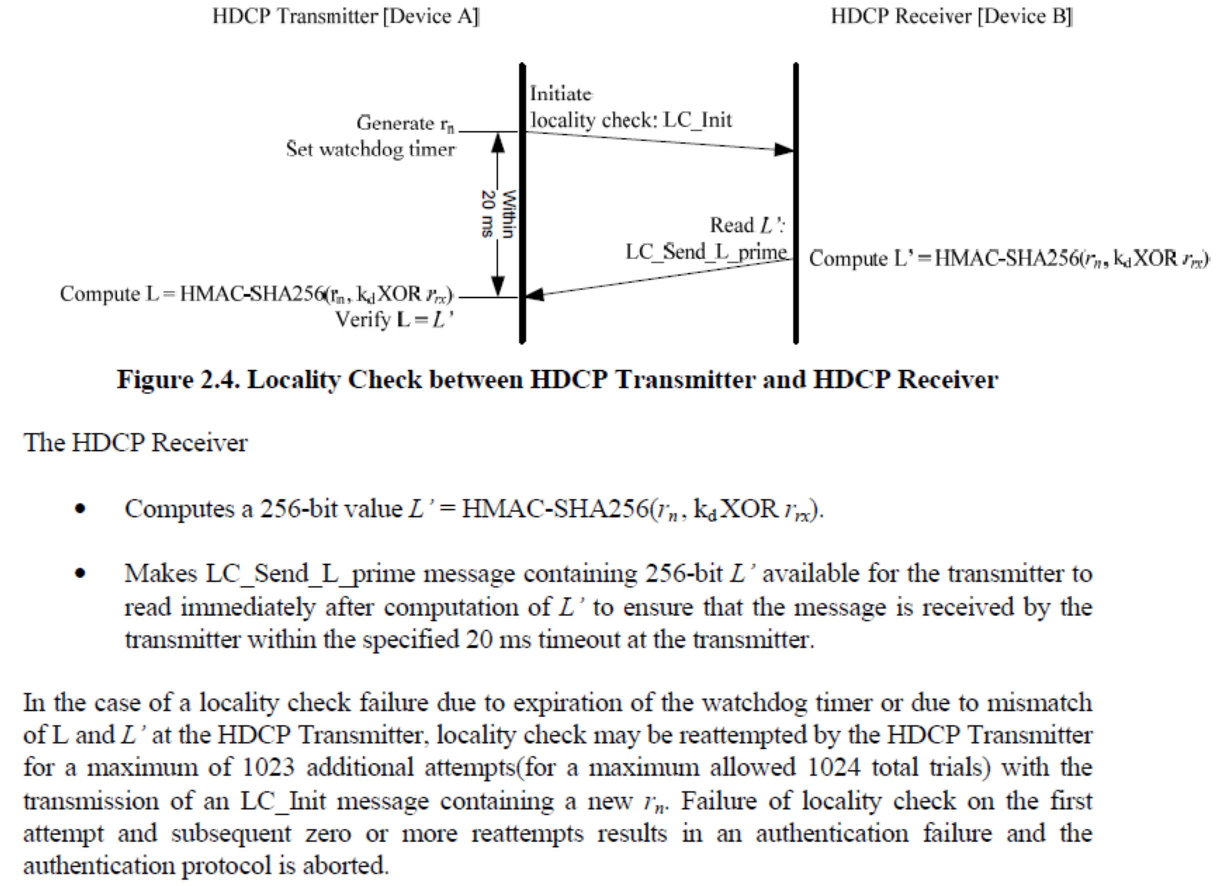


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

The second signal, *e.g.*, L', is derived from a secret.

The value of L' is derived from k_d.

Compute L' = HMAC-SHA256(r_n, k_dXOR r_{rx})

Id.

The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

> Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0 \| dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 14-15.

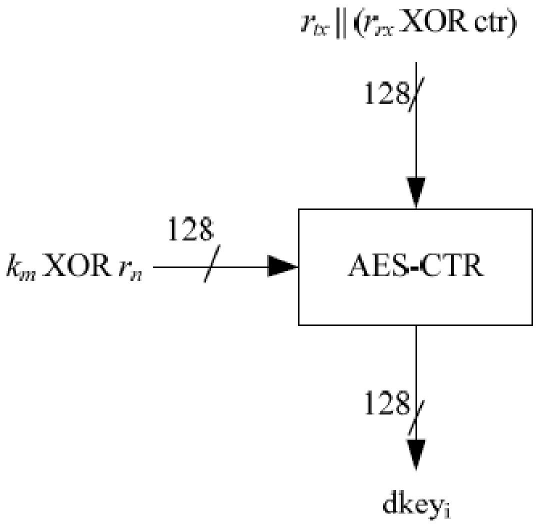


Figure 2.10. Key Derivation

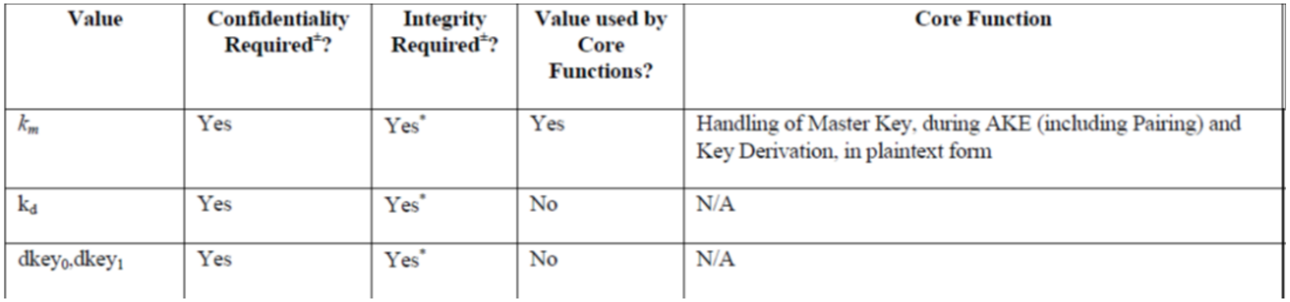
Id. at 25.

> **Master Key**. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver.

Id. at 8.

Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"



Value	Confidentiality Required$^{\pm}$?	Integrity Required$^{\pm}$?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
$dkey_0, dkey_1$	Yes	Yes*	No	N/A

Id. at 67 (abridged).

The Accused Product provides the Master Key, k_m, encrypted to the second device irrespective of whether the Accused Product previously stored a k_m corresponding to the second device.

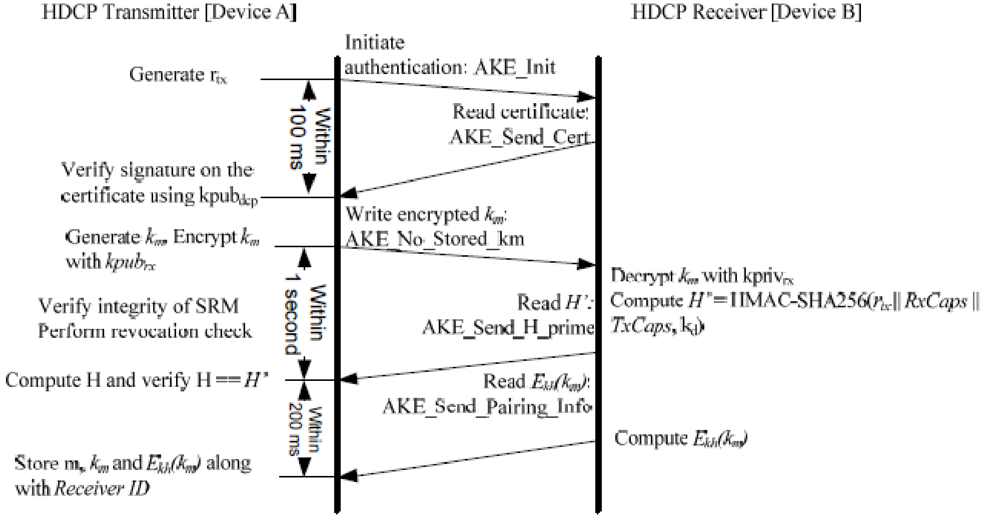


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

- Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.

Id. at 13.

- If AKE_No_Stored_km is received, the HDCP Receiver
  - Decrypts k_m with kpriv_{rx} using RSAES-OAEP decryption scheme.
  - Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. $k_d = \text{dkey}_0 \| \text{dkey}_1$, where dkey_0 and dkey_1 are derived keys generated when ctr = 0 and ctr = 1 respectively. dkey_0 and dkey_1 are in big-endian order.

Id. at 14.

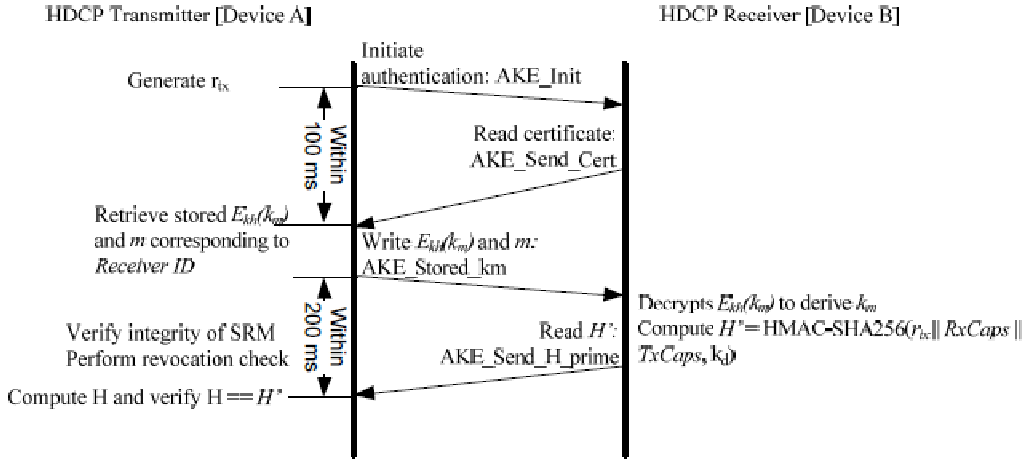
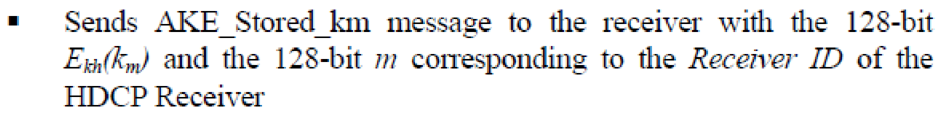


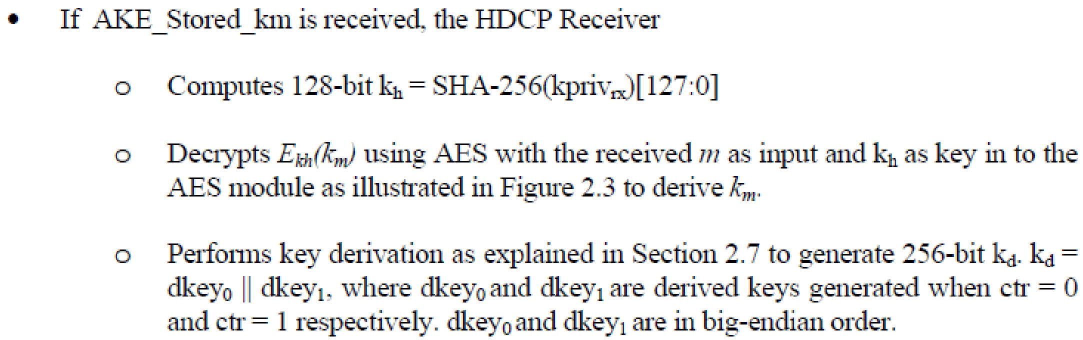
Figure 2.2. Authentication and Key Exchange (With Stored k_m)

Id. at 12.



- Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit *m* corresponding to the *Receiver ID* of the HDCP Receiver

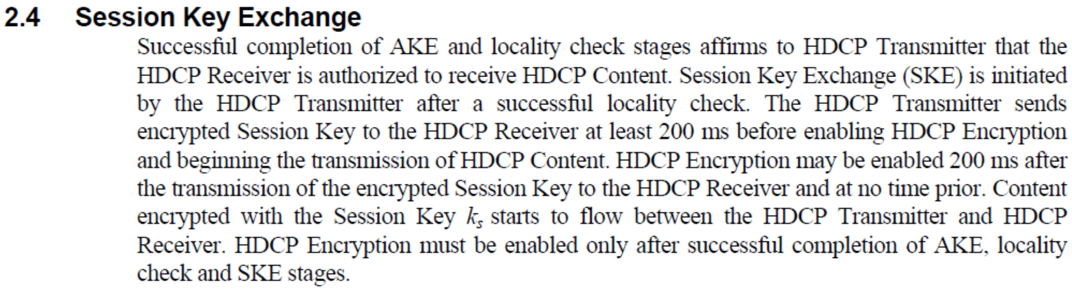
Id. at 14.



- If AKE_Stored_km is received, the HDCP Receiver
  - Computes 128-bit k_h = SHA-256($kpriv_{rx}$)[127:0]
  - Decrypts $E_{kh}(k_m)$ using AES with the received *m* as input and k_h as key in to the AES module as illustrated in Figure 2.3 to derive k_m.
  - Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 15.

The Accused Product proceeds to session key exchange and providing of the protected content to the second device after successful completion of the AKE stage and Locality Check.

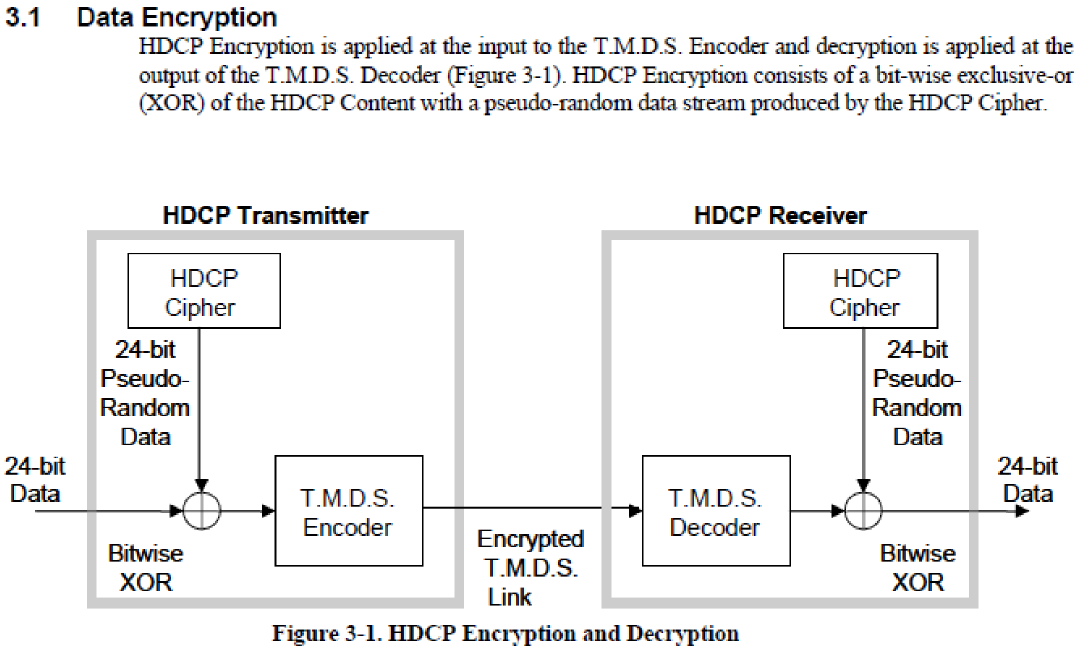


2.4 Session Key Exchange

Successful completion of AKE and locality check stages affirms to HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. Session Key Exchange (SKE) is initiated by the HDCP Transmitter after a successful locality check. The HDCP Transmitter sends encrypted Session Key to the HDCP Receiver at least 200 ms before enabling HDCP Encryption and beginning the transmission of HDCP Content. HDCP Encryption may be enabled 200 ms after the transmission of the encrypted Session Key to the HDCP Receiver and at no time prior. Content encrypted with the Session Key k_s starts to flow between the HDCP Transmitter and HDCP Receiver. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages.

Id. at 17.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

	 **3.1 Data Encryption** HDCP Encryption is applied at the input to the T.M.D.S. Encoder and decryption is applied at the output of the T.M.D.S. Decoder (Figure 3-1). HDCP Encryption consists of a bit-wise exclusive-or (XOR) of the HDCP Content with a pseudo-random data stream produced by the HDCP Cipher. **Figure 3-1. HDCP Encryption and Decryption** *Id.* at 50. *See also*:

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

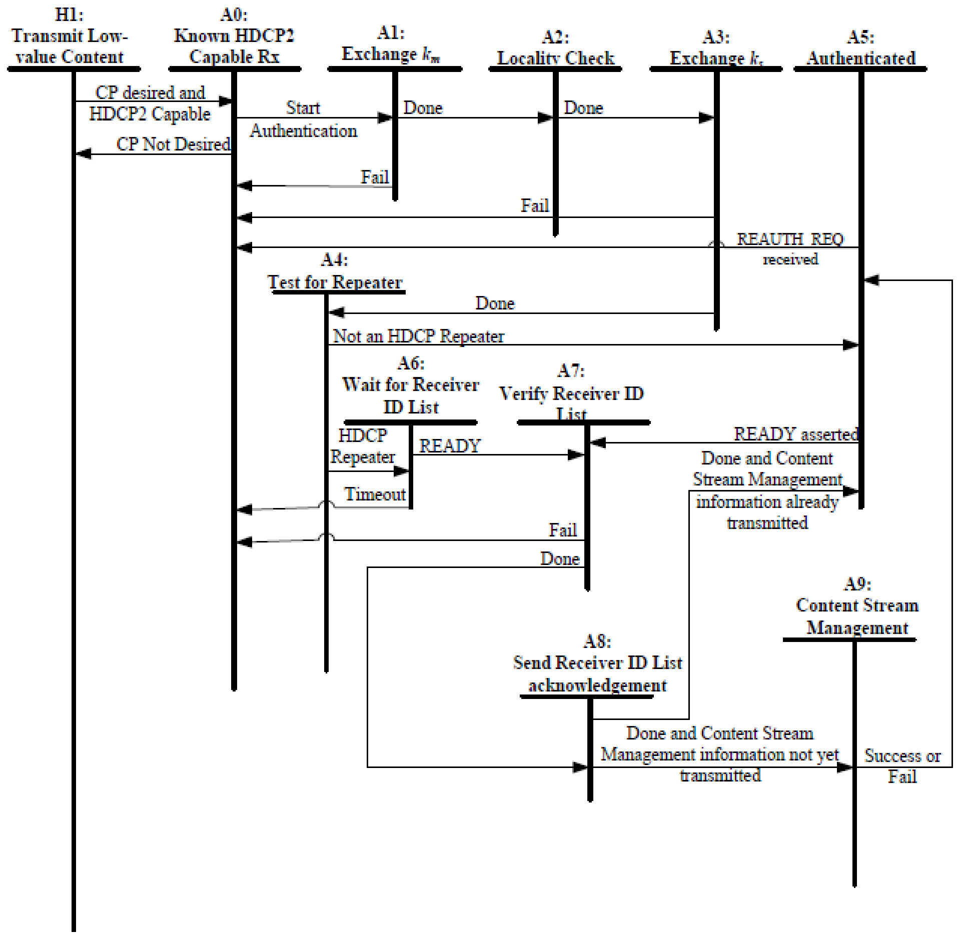
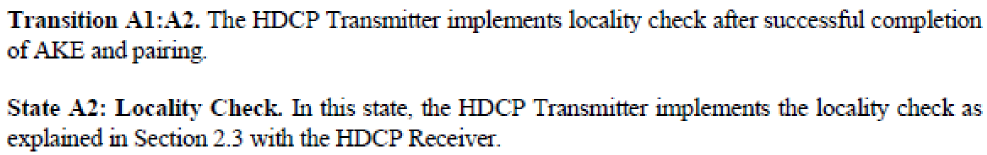


Figure 2.12. HDCP Transmitter Authentication Protocol State Diagram

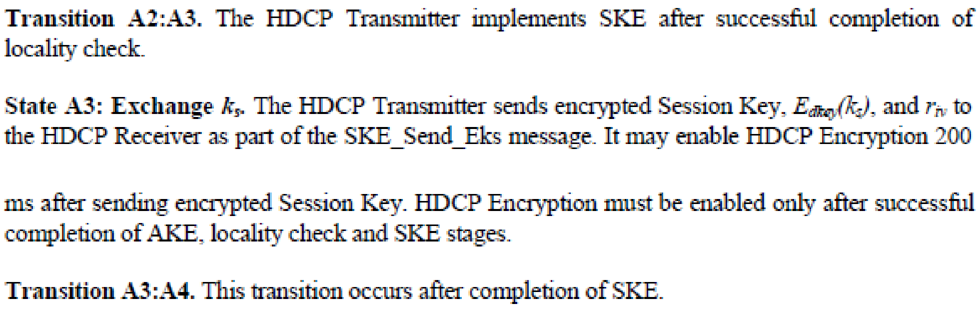
Id. at 27.



Transition A1:A2. The HDCP Transmitter implements locality check after successful completion of AKE and pairing.

State A2: Locality Check. In this state, the HDCP Transmitter implements the locality check as explained in Section 2.3 with the HDCP Receiver.

"provide the protected content to the second device when the second signal is derived from a secret and a time between the providing of the first signal and the receiving of the second signal is less than a predetermined time,"

	Id. at 28.  **Transition A2:A3.** The HDCP Transmitter implements SKE after successful completion of locality check. **State A3: Exchange k_s.** The HDCP Transmitter sends encrypted Session Key, $E_{dkey}(k_s)$, and r_{iv} to the HDCP Receiver as part of the SKE_Send_Eks message. It may enable HDCP Encryption 200 ms after sending encrypted Session Key. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages. **Transition A3:A4.** This transition occurs after completion of SKE. *Id.* at 28-29.

"wherein the secret is known by the first device."

<table>
<tr>
<td>wherein the secret is known by the first device.</td>
<td>

The secret is known by the first device. For example, the Accused Product generates and/or stores the Master Key, k_m, a secret, and thus knows k_m.

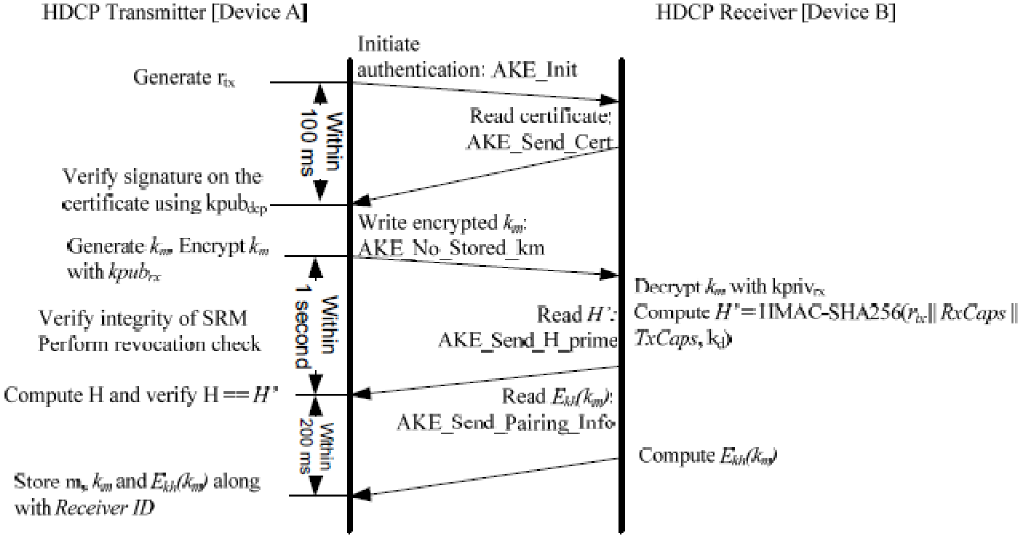
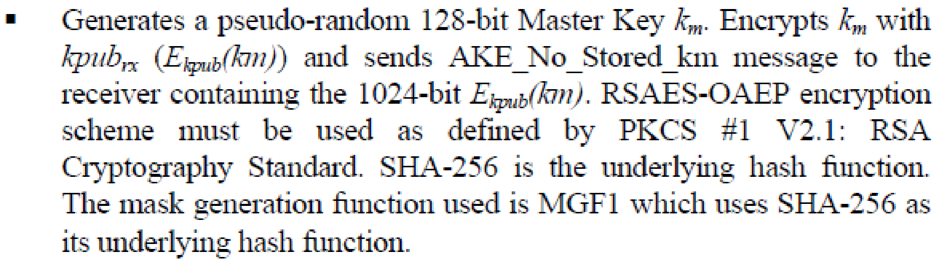


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

HDMI HDCP 2.2 at 12.



- Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.

Id. at 13.

</td>
</tr>
</table>

"wherein the secret is known by the first device."

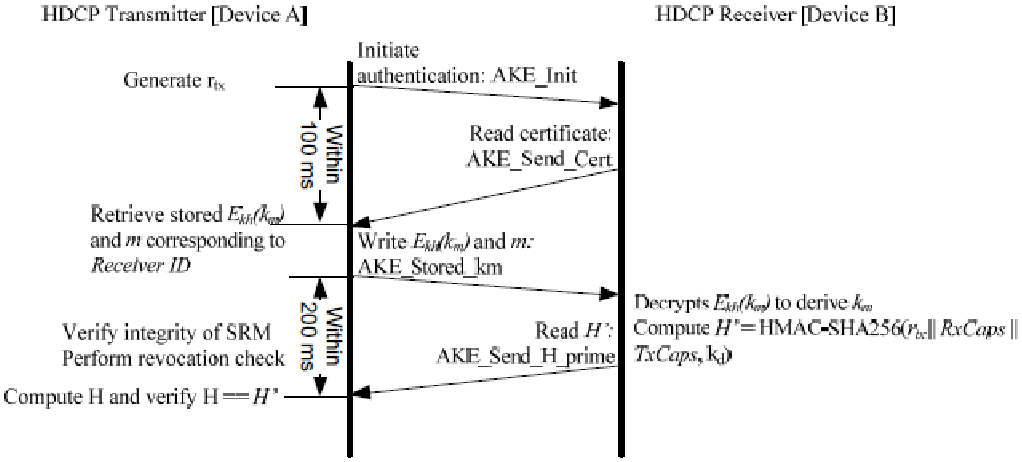
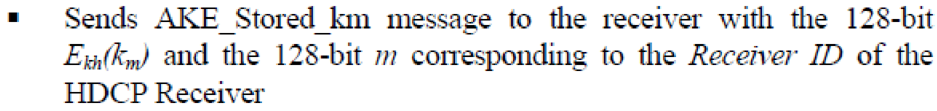


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

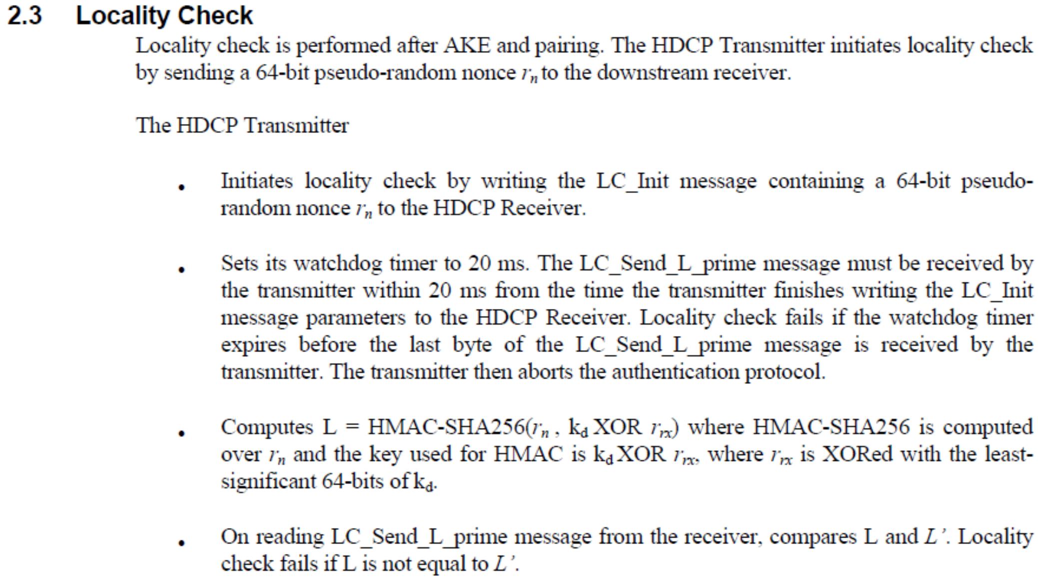
Id. at 12.



- Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit m corresponding to the *Receiver ID* of the HDCP Receiver

Id. at 14.

The Accused Product also knows k_d, which is a secret.



2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
- Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol.
- Computes L = HMAC-SHA256(r_n, k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d.
- On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'.

Id. at 16.

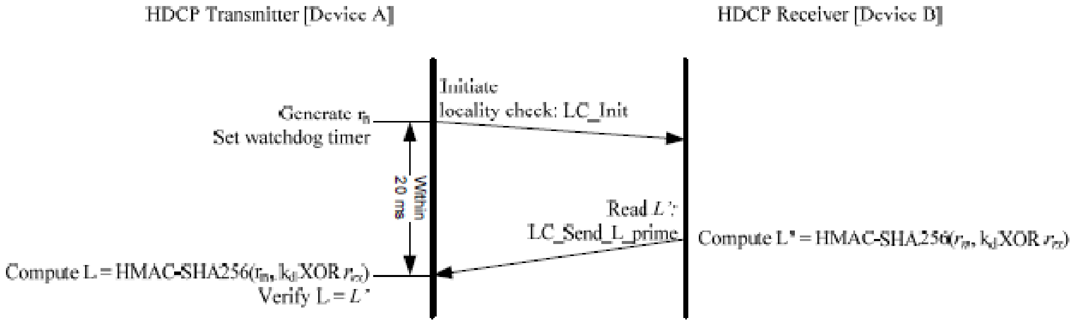
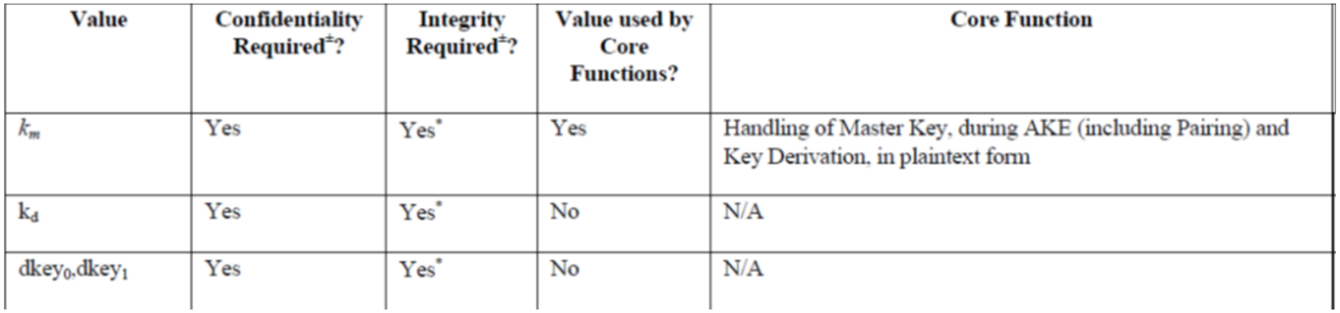


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

Id. at 17.



Value	Confidentiality Required±?	Integrity Required±?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
$dkey_0$,$dkey_1$	Yes	Yes*	No	N/A

Id. at 67 (abridged).

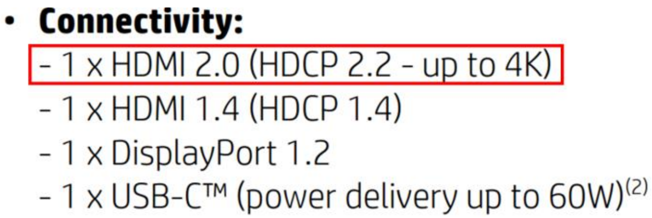
EXHIBIT F

U.S. Patent No. 10,298,564

HP Product / MediaTek Product

	
HP ENVY 27 27-inch Monitor (Part # W5A12AA) (“HP Product” or “Accused Product”)	MediaTek video processing system and components thereof including MStar MST9U11H1 Processor, main board hardware, integrated operating system, middleware, application program, video processing, and/or digital rights management (“DRM”) software that runs on the HP Product (“MediaTek Product” or “Accused Product”)

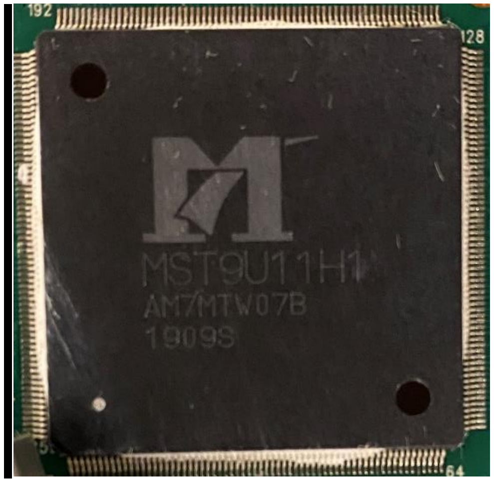
"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:	Each of the HP Product and the MediaTek Product is a second device for receiving delivery of a protected content from a first device, the processor circuit arranged to execute instructions, and is referred to herein as an “Accused Product.” For example, the HP Product is an HDMI receiver with HDCP 2.2 for receiving delivery of protected content from another device, such as an HDMI transmitter with HDCP 2.2.  HP, HP ENVY 27 Display, https://www8.hp.com/h20195/v2/GetDocument.aspx?docname=4aa7-5247enuc. The HP Product includes an HDMI 2.0 (HDCP 2.2 – up to 4K) port.  • **Connectivity:** - 1 x HDMI 2.0 (HDCP 2.2 - up to 4K) - 1 x HDMI 1.4 (HDCP 1.4) - 1 x DisplayPort 1.2 - 1 x USB-C™ (power delivery up to 60W)(2) *Id.*

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

HP Product Image.

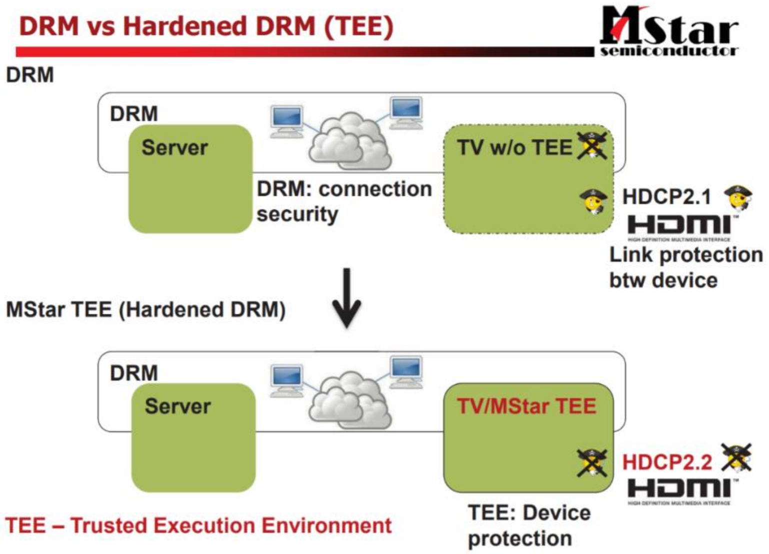
The HP Product comprises a processor circuit, the processor circuit arranged to execute instructions as set forth in the body of the claim. The HP Product includes the MStar MST9U11H1 SoC (the “MStar SoC”).

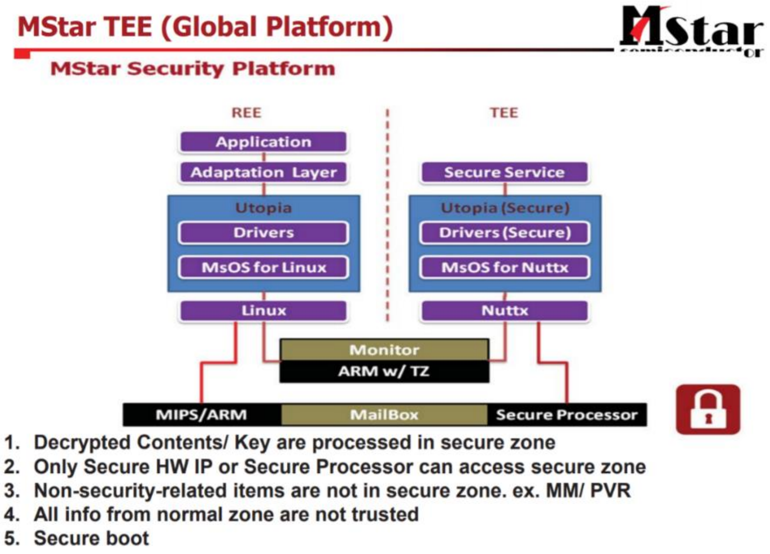


Accused Product Teardown (SoC).

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

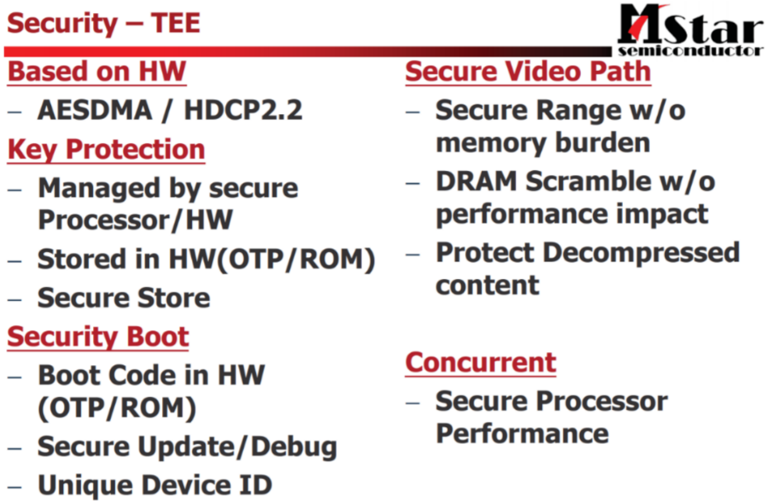
<table>
<tr>
<td></td>
<td>
The MStar SoC implements “Hardened DRM” – Mstar Trusted Execution Environment (TEE) that includes hardware support for HDCP 2.2.





1. Decrypted Contents/ Key are processed in secure zone
2. Only Secure HW IP or Secure Processor can access secure zone
3. Non-security-related items are not in secure zone. ex. MM/ PVR
4. All info from normal zone are not trusted
5. Secure boot
</td>
</tr>
</table>

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

<table>
<tr>
<td></td>
<td>


Security – TEE

MStar semiconductor

Based on HW

- AESDMA / HDCP2.2

Key Protection

- Managed by secure Processor/HW
- Stored in HW(OTP/ROM)
- Secure Store

Security Boot

- Boot Code in HW (OTP/ROM)
- Secure Update/Debug
- Unique Device ID

Secure Video Path

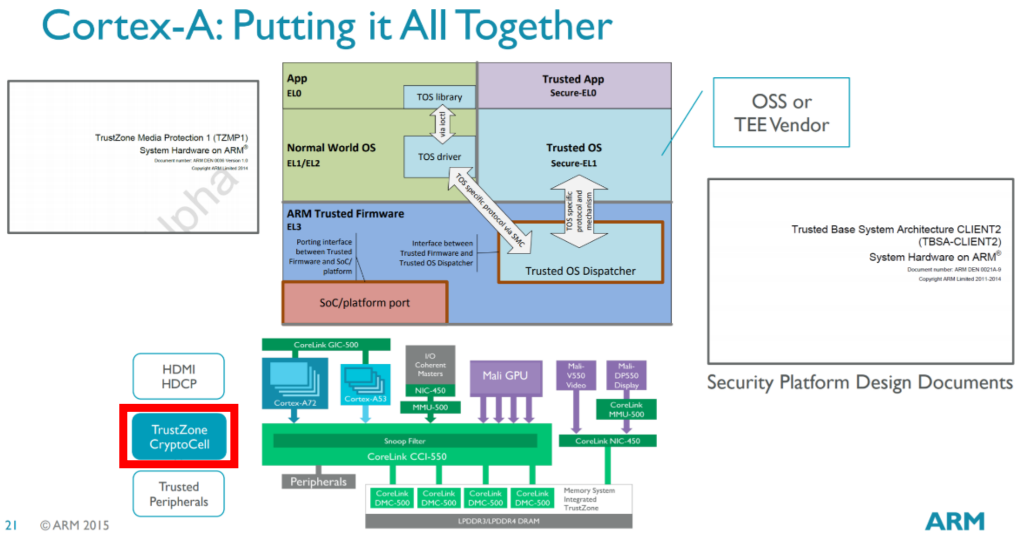
- Secure Range w/o memory burden
- DRAM Scramble w/o performance impact
- Protect Decompressed content

Concurrent

- Secure Processor Performance

MStar Semiconductor, Security Evolution on TV/OTT (Jun. 2015), *available at* https://ecfsapi.fcc.gov/file/60001077389.pdf, at pp. 2, 5, 6.

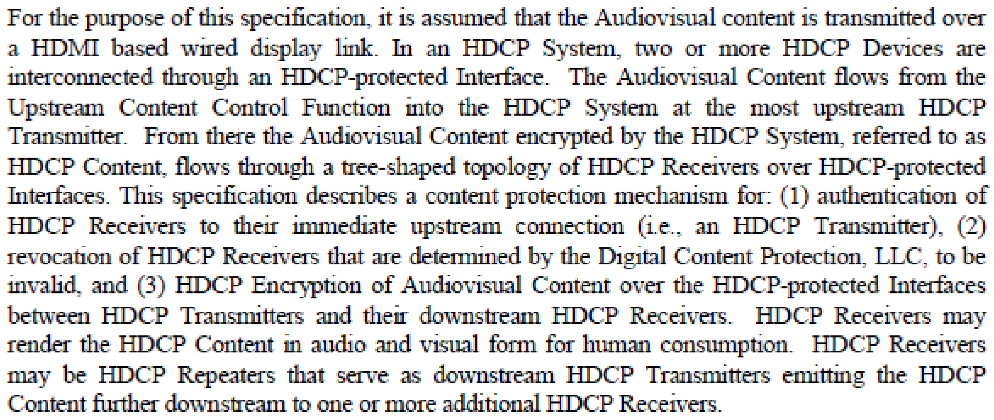
On information and belief, the MStar SoC includes an ARM TrustZone-enabled Cortex-A CPU and Mali GPU, which provide hardware support for HDCP 2.2 and execute instructions.



Cortex-A: Putting it All Together

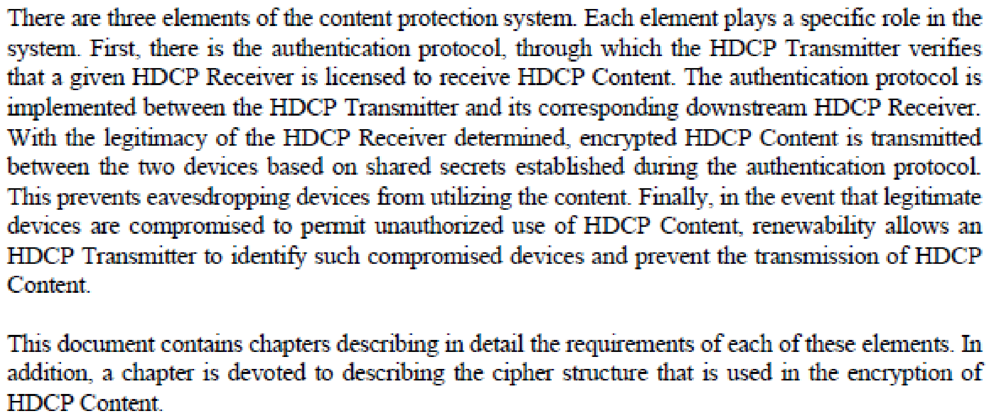
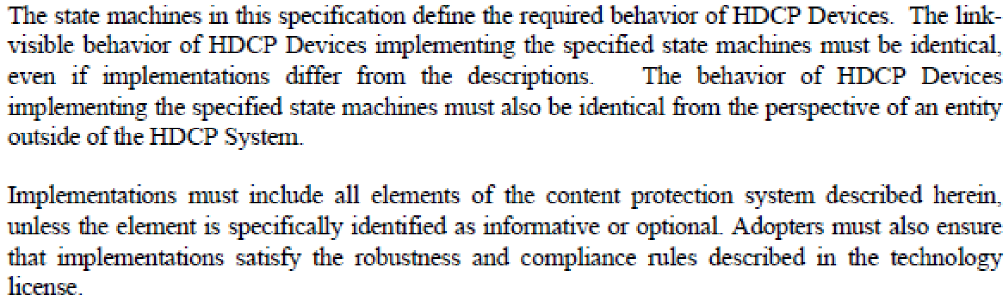
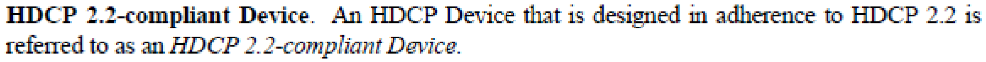
21 © ARM 2015
</td>
</tr>
</table>

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

	ARM, Designing Security & Trust into Connected Devices (Nov. 10, 2015), *available at* https://community.arm.com/cfs-file/__key/telligent-evolution-components-attachments/01-2142-00-00-00-00-67-58/ARM-Techcon-Security-2015.pdf, at p. 21. Upon information and belief, the Accused Product is compliant with the High-bandwidth Digital Content Protection System Revision 2.2 ("HDCP 2.2") protocol. The Accused Product supports HDCP 2.2 for protecting content between devices.  For the purpose of this specification, it is assumed that the Audiovisual content is transmitted over a HDMI based wired display link. In an HDCP System, two or more HDCP Devices are interconnected through an HDCP-protected Interface. The Audiovisual Content flows from the Upstream Content Control Function into the HDCP System at the most upstream HDCP Transmitter. From there the Audiovisual Content encrypted by the HDCP System, referred to as HDCP Content, flows through a tree-shaped topology of HDCP Receivers over HDCP-protected Interfaces. This specification describes a content protection mechanism for: (1) authentication of HDCP Receivers to their immediate upstream connection (i.e., an HDCP Transmitter), (2) revocation of HDCP Receivers that are determined by the Digital Content Protection, LLC, to be invalid, and (3) HDCP Encryption of Audiovisual Content over the HDCP-protected Interfaces between HDCP Transmitters and their downstream HDCP Receivers. HDCP Receivers may render the HDCP Content in audio and visual form for human consumption. HDCP Receivers may be HDCP Repeaters that serve as downstream HDCP Transmitters emitting the HDCP Content further downstream to one or more additional HDCP Receivers. High-bandwidth Digital Content Protection System Mapping HDCP to HDMI, Rev. 2.2 (Feb. 13, 2013), *available at* https://www.digital-cp.com/sites/default/files/specifications/HDCP%20on%20HDMI%20Specification%20Rev2_2_Final1.pdf ("HDMI HDCP 2.2") at 5.

U.S. Patent No. 10,298,564: Claim 1

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

	 There are three elements of the content protection system. Each element plays a specific role in the system. First, there is the authentication protocol, through which the HDCP Transmitter verifies that a given HDCP Receiver is licensed to receive HDCP Content. The authentication protocol is implemented between the HDCP Transmitter and its corresponding downstream HDCP Receiver. With the legitimacy of the HDCP Receiver determined, encrypted HDCP Content is transmitted between the two devices based on shared secrets established during the authentication protocol. This prevents eavesdropping devices from utilizing the content. Finally, in the event that legitimate devices are compromised to permit unauthorized use of HDCP Content, renewability allows an HDCP Transmitter to identify such compromised devices and prevent the transmission of HDCP Content. This document contains chapters describing in detail the requirements of each of these elements. In addition, a chapter is devoted to describing the cipher structure that is used in the encryption of HDCP Content. *Id*. at 9. The Accused Product is an HDCP Device, and more specifically an HDCP 2.2-compliant Device, capable of functioning as an HDCP Receiver and that implements required functionality of HDMI HDCP 2.2 including the functions required by the HDCP Receiver State Diagram.  The state machines in this specification define the required behavior of HDCP Devices. The link-visible behavior of HDCP Devices implementing the specified state machines must be identical, even if implementations differ from the descriptions. The behavior of HDCP Devices implementing the specified state machines must also be identical from the perspective of an entity outside of the HDCP System. Implementations must include all elements of the content protection system described herein, unless the element is specifically identified as informative or optional. Adopters must also ensure that implementations satisfy the robustness and compliance rules described in the technology license. *Id*. at 5.  **HDCP 2.2-compliant Device**. An HDCP Device that is designed in adherence to HDCP 2.2 is referred to as an *HDCP 2.2-compliant Device*. *Id*. at 6.

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

> **HDCP Device**. Any device that contains one or more HDCP-protected Interface Port and is designed in adherence to HDCP is referred to as an *HDCP Device*.

Id. at 7.

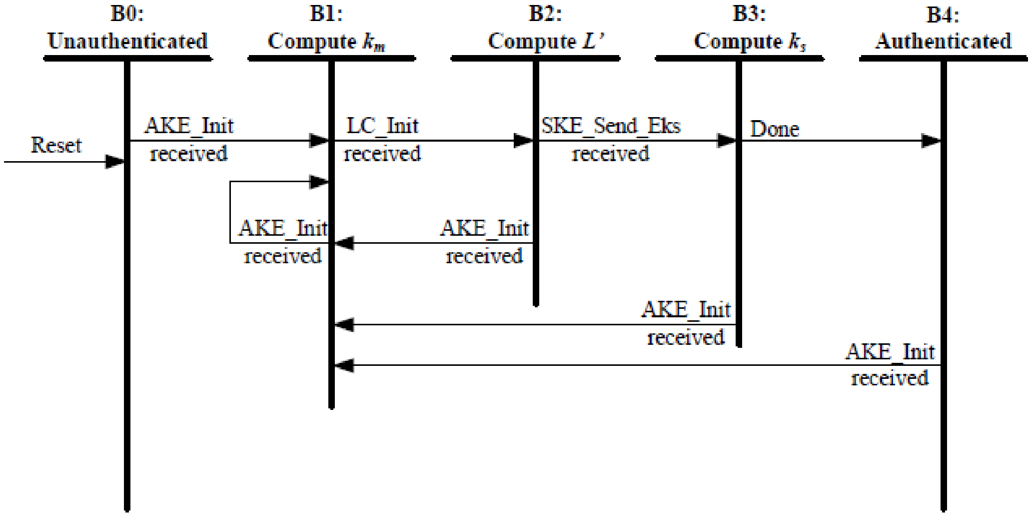


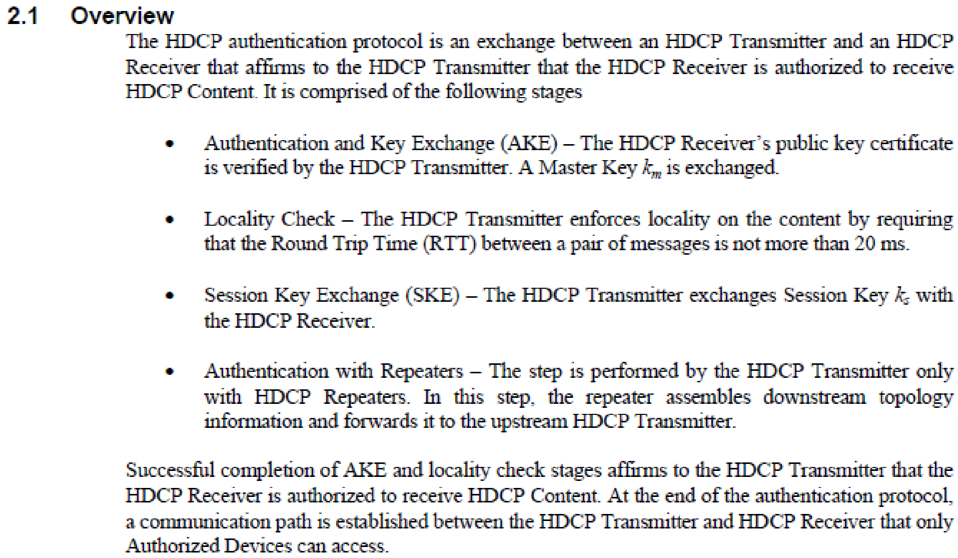
Figure 2.13. HDCP Receiver Authentication Protocol State Diagram

Id. at 31-32.

The Accused Product includes, for example, a bit in its HDCP2Version register identifying the Accused Product as HDCP 2 capable.

> **State H1: Transmit Low-value Content.** In this state, the transmitter reads the HDCP2Version register. The transmitter determines that the receiver is HDCP 2 capable by reading bit[2] in the receiver's HDCP2Version register. If this bit is set to 1, it indicates that the receiver is HDCP 2 capable. In this state the transmitter should begin sending an unencrypted signal with HDCP Encryption disabled. The transmitted signal can be a low value content or informative on-screen display. This will ensure that a valid video signal is displayed to the user before and during authentication.

"1. A second device for receiving delivery of a protected content from a first device, the second device comprising a processor circuit, the processor circuit arranged to execute instructions, the instructions arranged to:"

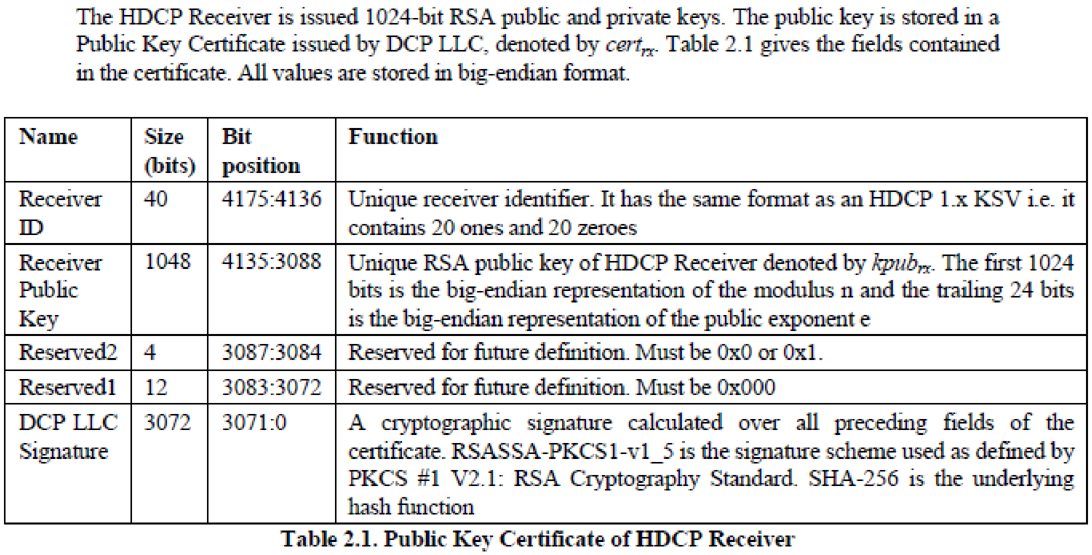
	Id. at 27. The Accused Product receives delivery of protected content from a first device.  **2.1 Overview** The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. *Id*. at 11.

"provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;"

<table>
<tr>
<td>provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;</td>
<td>

The instructions of the Accused Product are arranged to provide a certificate, *e.g.*, $cert_{rx}$, to the first device (transmitter) as part of the Authentication and Key Exchange (AKE) stage of the HDCP 2.2 protocol and prior to receiving a first signal, *e.g.*, the LC_Init message including r_n, wherein the first signal is sent by the first device, and wherein the certificate is associated with the Accused Product (second device).

The certificate, $cert_{rx}$, includes a Receiver ID for the Accused Product, Receiver Public Key, and a cryptographic signature, amongst other information.

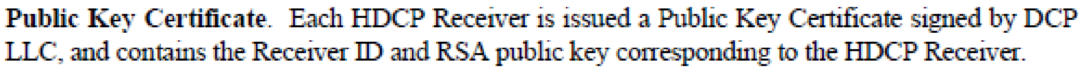


The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.

Name	**Size (bits)**	**Bit position**	**Function**
Receiver ID	40	4175:4136	Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes
Receiver Public Key	1048	4135:3088	Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e
Reserved2	4	3087:3084	Reserved for future definition. Must be 0x0 or 0x1.
Reserved1	12	3083:3072	Reserved for future definition. Must be 0x000
DCP LLC Signature	3072	3071:0	A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function

Table 2.1. Public Key Certificate of HDCP Receiver

HDMI HDCP 2.2 at 11.



Public Key Certificate. Each HDCP Receiver is issued a Public Key Certificate signed by DCP LLC, and contains the Receiver ID and RSA public key corresponding to the HDCP Receiver.

Id. at 8.

The Accused Product provides the certificate to the transmitter as part of the AKE stage, irrespective of whether the transmitter has a Master Key k_m stored corresponding to the Receiver ID.

</td>
</tr>
</table>

"provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;"

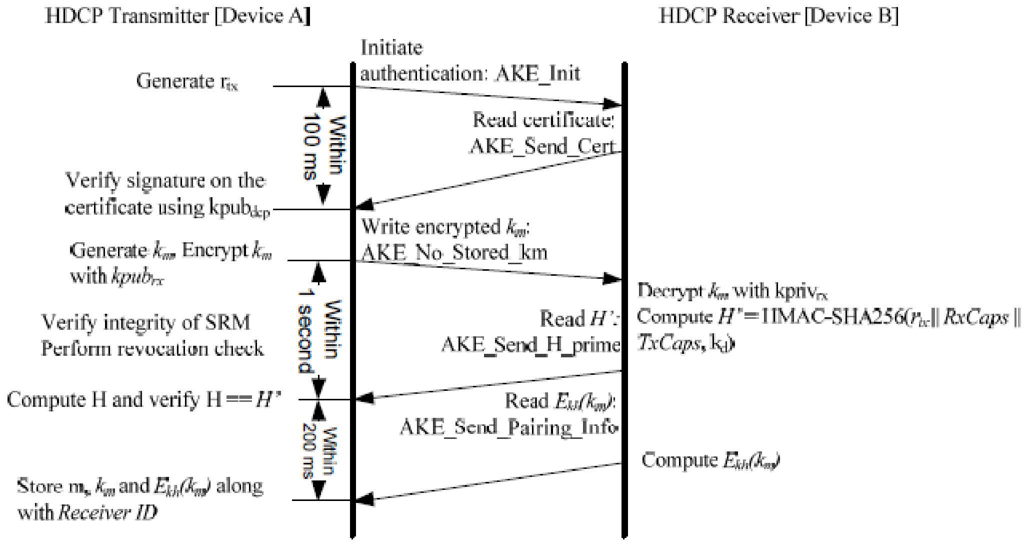


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

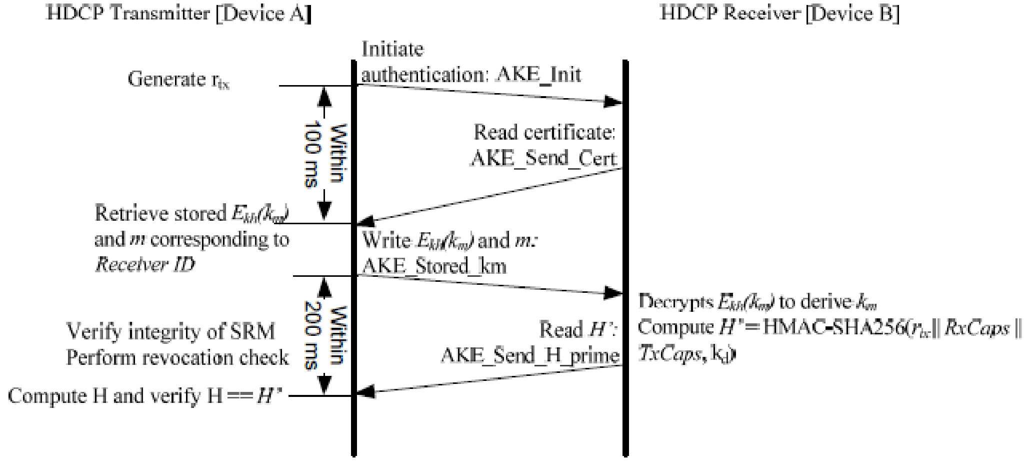
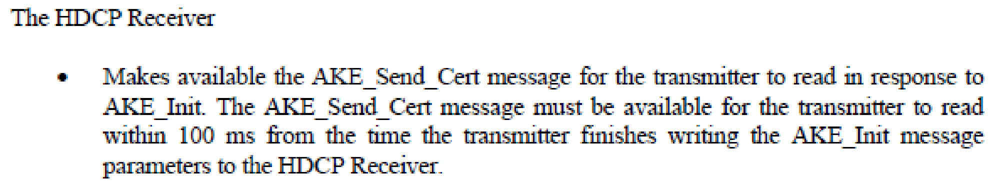


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

Id.

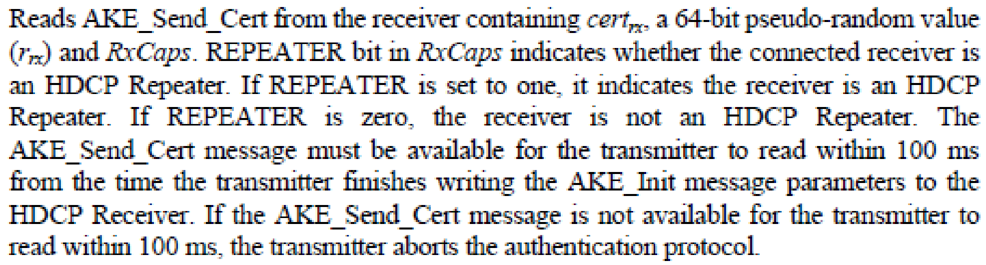
The Accused Product provides the certificate to the first device as part of the AKE_Send_Cert message.



The HDCP Receiver

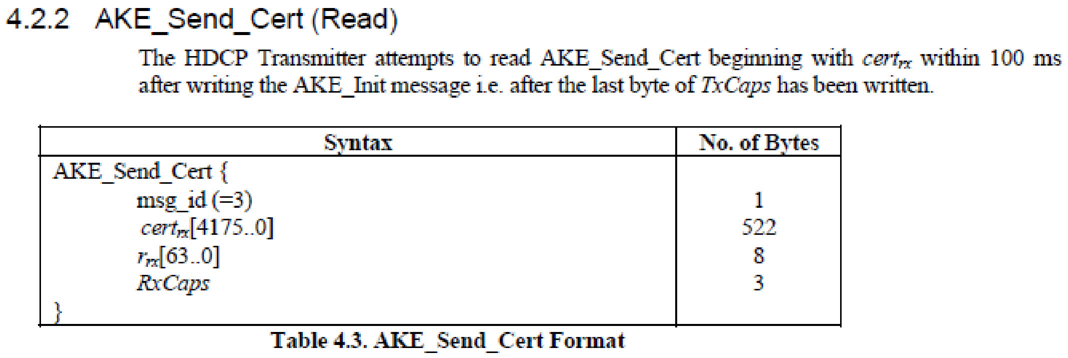
- Makes available the AKE_Send_Cert message for the transmitter to read in response to AKE_Init. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver.

Id. at 14.



Reads AKE_Send_Cert from the receiver containing $cert_{rx}$, a 64-bit pseudo-random value (r_{rx}) and *RxCaps*. REPEATER bit in *RxCaps* indicates whether the connected receiver is an HDCP Repeater. If REPEATER is set to one, it indicates the receiver is an HDCP Repeater. If REPEATER is zero, the receiver is not an HDCP Repeater. The AKE_Send_Cert message must be available for the transmitter to read within 100 ms from the time the transmitter finishes writing the AKE_Init message parameters to the HDCP Receiver. If the AKE_Send_Cert message is not available for the transmitter to read within 100 ms, the transmitter aborts the authentication protocol.

Id. at 13.



4.2.2 AKE_Send_Cert (Read)

The HDCP Transmitter attempts to read AKE_Send_Cert beginning with $cert_{rx}$ within 100 ms after writing the AKE_Init message i.e. after the last byte of *TxCaps* has been written.

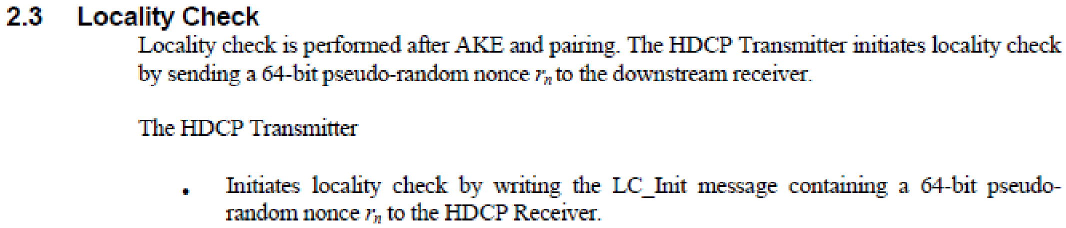
Syntax	No. of Bytes
AKE_Send_Cert {	
msg_id (=3)	1
$cert_{rx}$[4175..0]	522
r_{rx}[63..0]	8
RxCaps	3
}	

Table 4.3. AKE_Send_Cert Format

Id. at 57.

"provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;"

The Accused Product provides the certificate to the first device during the AKE stage prior to receiving the first signal, *e.g.*, the LC_Init message including r_n, as part of a Locality Check.



2.3 Locality Check

Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.

The HDCP Transmitter

- Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.

Id. at 16.

See also:

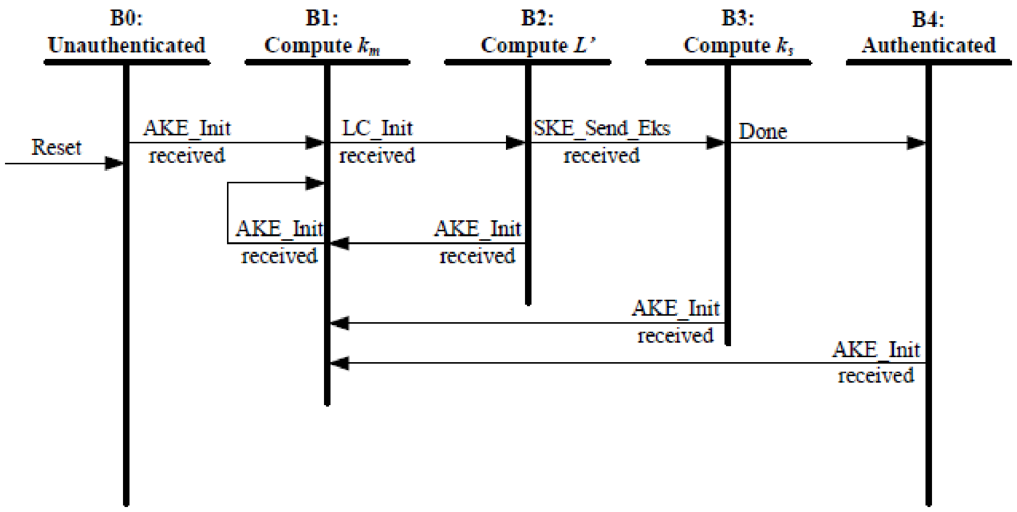
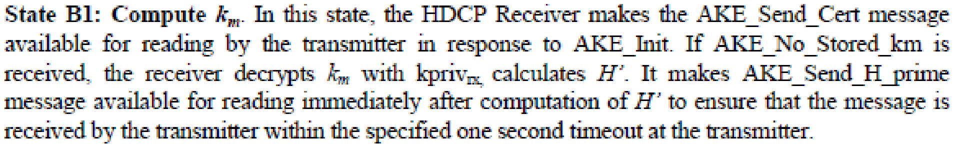


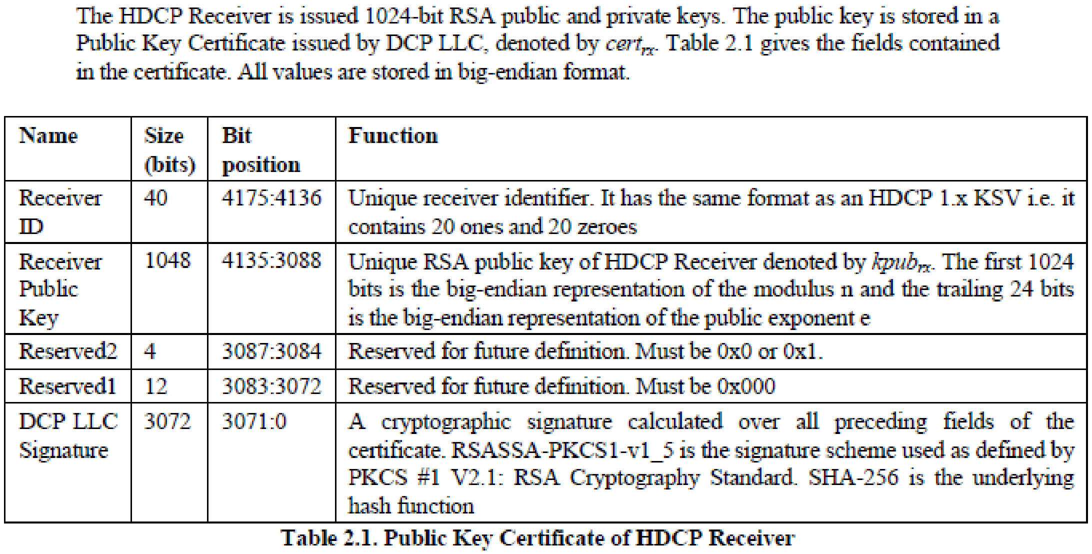
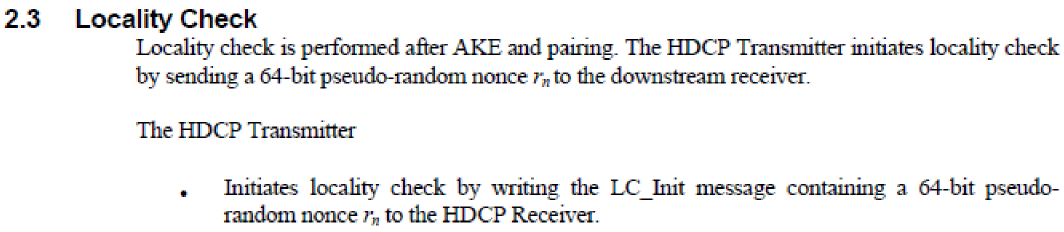
Figure 2.13. HDCP Receiver Authentication Protocol State Diagram

Id. at 31.

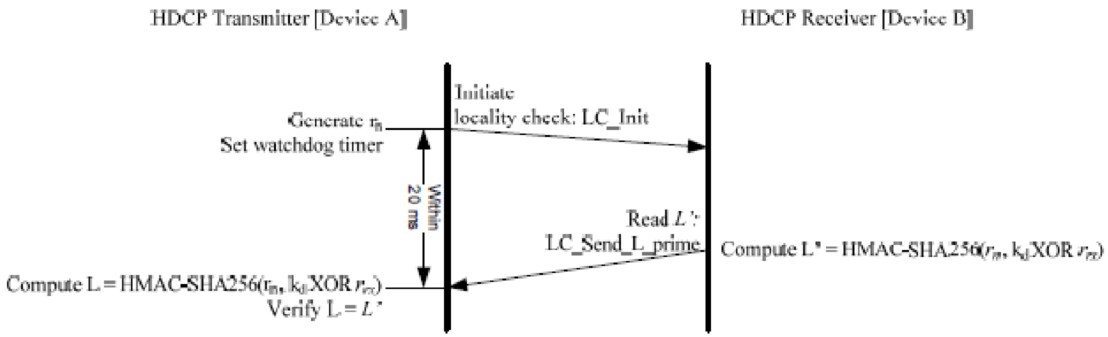
"provide a certificate to the first device prior to receiving a first signal, wherein the first signal is sent by the first device, wherein the certificate is associated with the second device;"

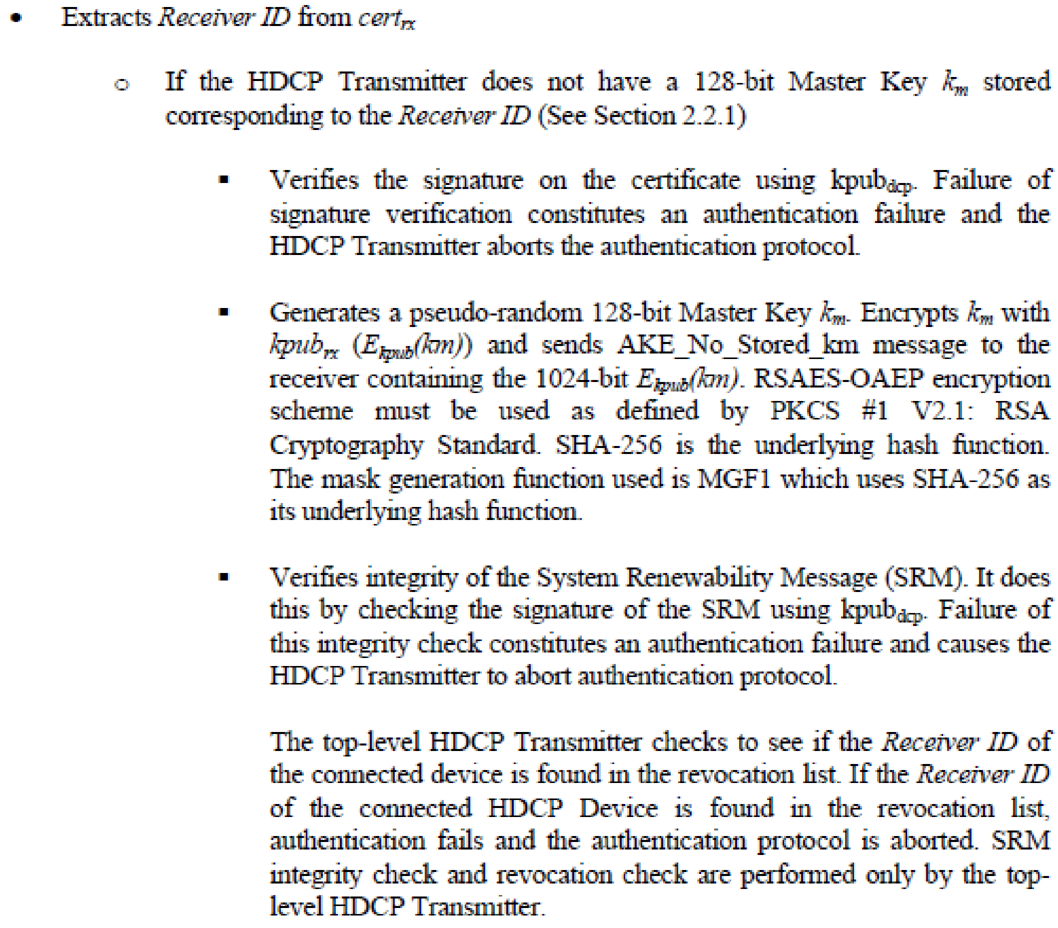
	 **State B1: Compute k_m.** In this state, the HDCP Receiver makes the AKE_Send_Cert message available for reading by the transmitter in response to AKE_Init. If AKE_No_Stored_km is received, the receiver decrypts k_m with kpriv$_{rx,}$ calculates H'. It makes AKE_Send_H_prime message available for reading immediately after computation of H' to ensure that the message is received by the transmitter within the specified one second timeout at the transmitter. *Id.*

"receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule;"

<table>
<tr>
<td>receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule;</td>
<td>
<p>The instructions of the Accused Product are arranged to receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule.</p>
<p>The Accused Product receives the LC_Init message including r_n when the certificate, $cert_{rx}$, indicates that the Accused Product is compliant with at least one compliance rule. For example, the certificate, $cert_{rx}$, includes a Receiver ID, Receiver Public Key, and a cryptographic signature.</p>

<p>The HDCP Receiver is issued 1024-bit RSA public and private keys. The public key is stored in a Public Key Certificate issued by DCP LLC, denoted by $cert_{rx}$. Table 2.1 gives the fields contained in the certificate. All values are stored in big-endian format.</p>
<table>
<tr><th>Name</th><th>Size (bits)</th><th>Bit position</th><th>Function</th></tr>
<tr><td>Receiver ID</td><td>40</td><td>4175:4136</td><td>Unique receiver identifier. It has the same format as an HDCP 1.x KSV i.e. it contains 20 ones and 20 zeroes</td></tr>
<tr><td>Receiver Public Key</td><td>1048</td><td>4135:3088</td><td>Unique RSA public key of HDCP Receiver denoted by $kpub_{rx}$. The first 1024 bits is the big-endian representation of the modulus n and the trailing 24 bits is the big-endian representation of the public exponent e</td></tr>
<tr><td>Reserved2</td><td>4</td><td>3087:3084</td><td>Reserved for future definition. Must be 0x0 or 0x1.</td></tr>
<tr><td>Reserved1</td><td>12</td><td>3083:3072</td><td>Reserved for future definition. Must be 0x000</td></tr>
<tr><td>DCP LLC Signature</td><td>3072</td><td>3071:0</td><td>A cryptographic signature calculated over all preceding fields of the certificate. RSASSA-PKCS1-v1_5 is the signature scheme used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function</td></tr>
</table>
<p>Table 2.1. Public Key Certificate of HDCP Receiver</p>
<p>HDMI HDCP 2.2 at 11.</p>

<p>2.3 Locality Check</p>
<p>Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver.</p>
<p>The HDCP Transmitter</p>
Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver.
<p><i>Id</i>. at 16.</p>
</td>
</tr>
</table>

"receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule;"

	 **Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver** *Id*. at 17. The Accused Product receives the LC_Init message including r_n when the certificate indicates, for example, that the certificate conforms with the format established by the HDCP specification (*see id*. at 11, excerpted above) and indicates a valid signature and a Receiver ID that is not in a revocation list.



- Extracts *Receiver ID* from $cert_{rx}$
  - If the HDCP Transmitter does not have a 128-bit Master Key k_m stored corresponding to the *Receiver ID* (See Section 2.2.1)
    - Verifies the signature on the certificate using kpub$_{dcp}$. Failure of signature verification constitutes an authentication failure and the HDCP Transmitter aborts the authentication protocol.
    - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function.
    - Verifies integrity of the System Renewability Message (SRM). It does this by checking the signature of the SRM using kpub$_{dcp}$. Failure of this integrity check constitutes an authentication failure and causes the HDCP Transmitter to abort authentication protocol.

      The top-level HDCP Transmitter checks to see if the *Receiver ID* of the connected device is found in the revocation list. If the *Receiver ID* of the connected HDCP Device is found in the revocation list, authentication fails and the authentication protocol is aborted. SRM integrity check and revocation check are performed only by the top-level HDCP Transmitter.

Id. at 13.

A valid signature in the certificate indicates, for example, that the second device is compliant with a set of compliance rules of the HDCP specification.

> 1.9 "**Compliance Rules**" means the technical requirements set out in Exhibit C, as such exhibit may be amended by Licensor from time to time in accordance with the terms of this Agreement.

"receive the first signal when the certificate indicates that the second device is compliant with at least one compliance rule;"

HDCP License Agreement, March 6, 2017, at 2, *available at* https://digital-cp.com/sites/default/files/HDCP%20License%20Agreement_March%206%2C%202017_FOR%20REVIEW%20ONLY.pdf.

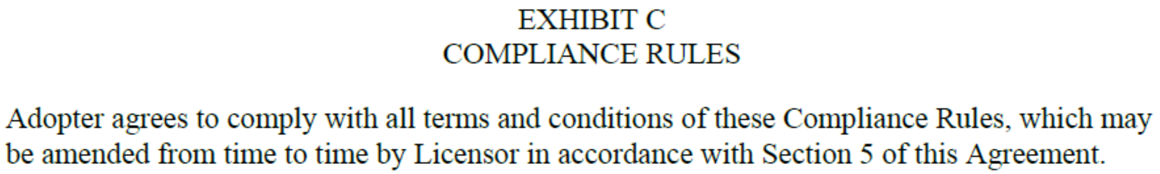


EXHIBIT C
COMPLIANCE RULES

Adopter agrees to comply with all terms and conditions of these Compliance Rules, which may be amended from time to time by Licensor in accordance with Section 5 of this Agreement.

Id. at Exhibit C.

See also:

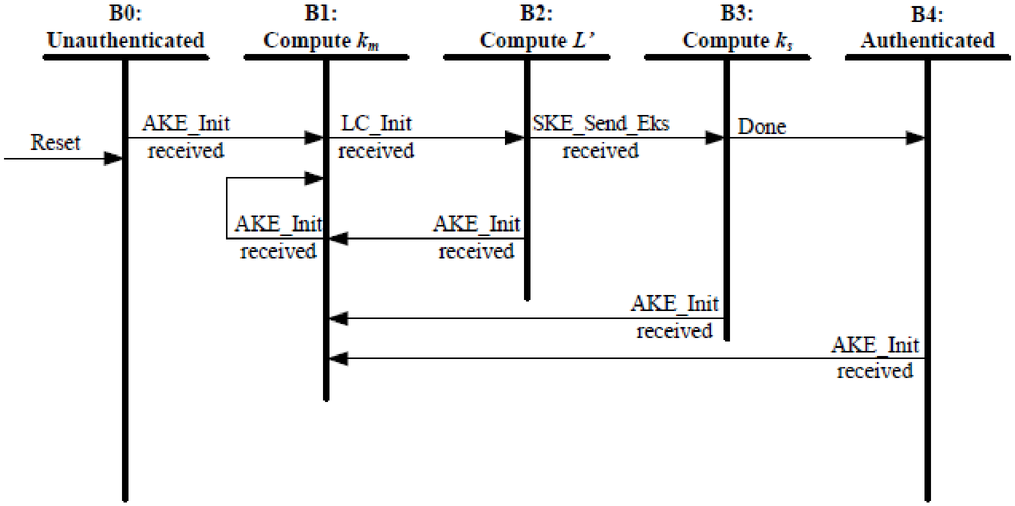
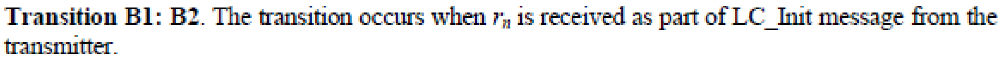


Figure 2.13. HDCP Receiver Authentication Protocol State Diagram

HDMI HDCP 2.2 at 31.



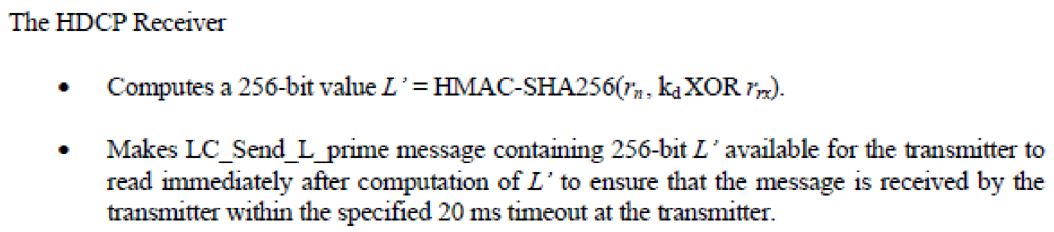
Transition B1: B2. The transition occurs when r_n is received as part of LC_Init message from the transmitter.

	Id.

"create a second signal, wherein the second signal is derived from a secret known by the second device;"

<table>
<tr>
<td>create a second signal, wherein the second signal is derived from a secret known by the second device;</td>
<td>

The instructions of the Accused Product are arranged to create a second signal, *e.g.*, L’.



The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

HDMI HDCP 2.2 at 17.

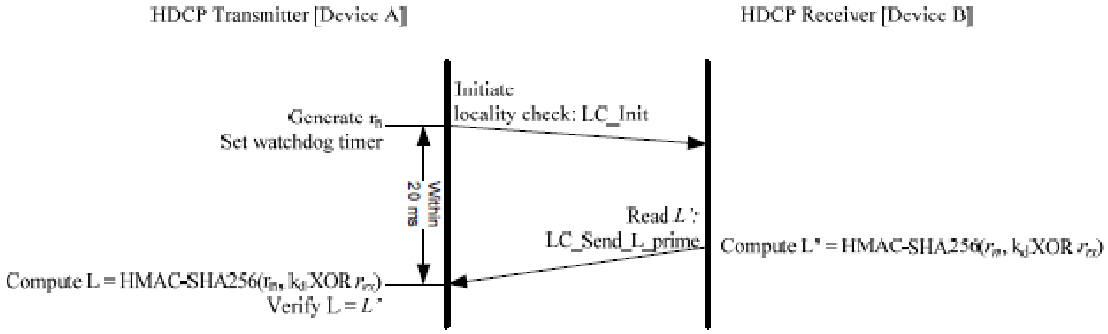


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

Id.

The second signal is derived from a secret known by the Accused Product (second device).

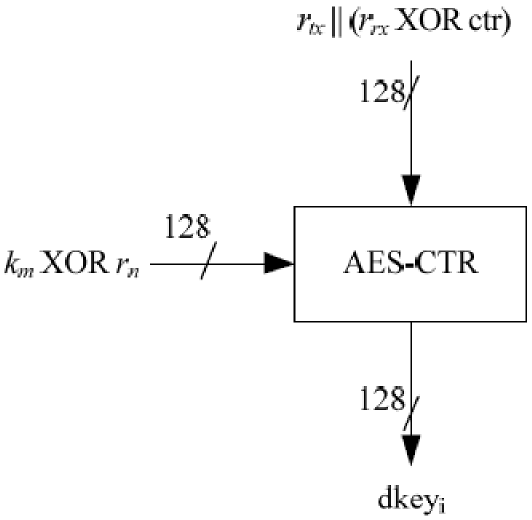
The value of L’ is derived from k_d.

Compute $L' = \text{HMAC-SHA256}(r_n, k_d \text{XOR } r_{rx})$

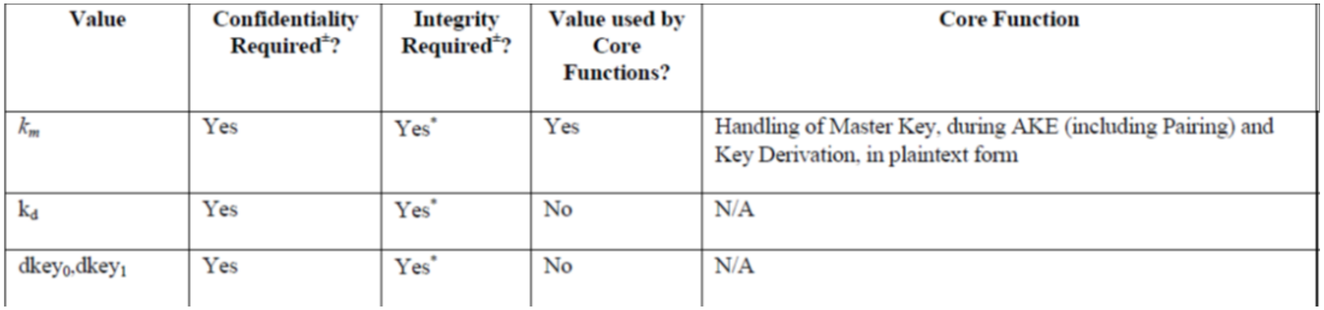
Id.

The value of k_d is based upon $dkey_0$ and $dkey_1$, each of which is derived from k_m, the Master Key.

</td>
</tr>
</table>

	Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. $k_d = dkey_0 \parallel dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order. *Id.* at 14-15.  **Figure 2.10. Key Derivation** *Id.* at 25. **Master Key**. A 128-bit random, secret cryptographic key negotiated between the HDCP Transmitter and the HDCP Receiver during Authentication and Key Exchange and used to pair the HDCP Transmitter with the HDCP Receiver. *Id.* at 8. Each of k_m, k_d, $dkey_0$ and $dkey_1$ is a secret known by the Accused Product.

"create a second signal, wherein the second signal is derived from a secret known by the second device;"



Value	Confidentiality Required‡?	Integrity Required‡?	Value used by Core Functions?	Core Function
k_m	Yes	Yes*	Yes	Handling of Master Key, during AKE (including Pairing) and Key Derivation, in plaintext form
k_d	Yes	Yes*	No	N/A
dkey$_0$,dkey$_1$	Yes	Yes*	No	N/A

Id. at 67 (abridged).

The Master Key, k_m, is received encrypted from the transmitter (first device) using the Accused Product's public key, kpub$_{rx}$. The Accused Product decrypts k_m using the Accused Product's private key, kpriv$_{rx}$, when the transmitter (first device) had not previously stored a k_m corresponding to the Accused Product.

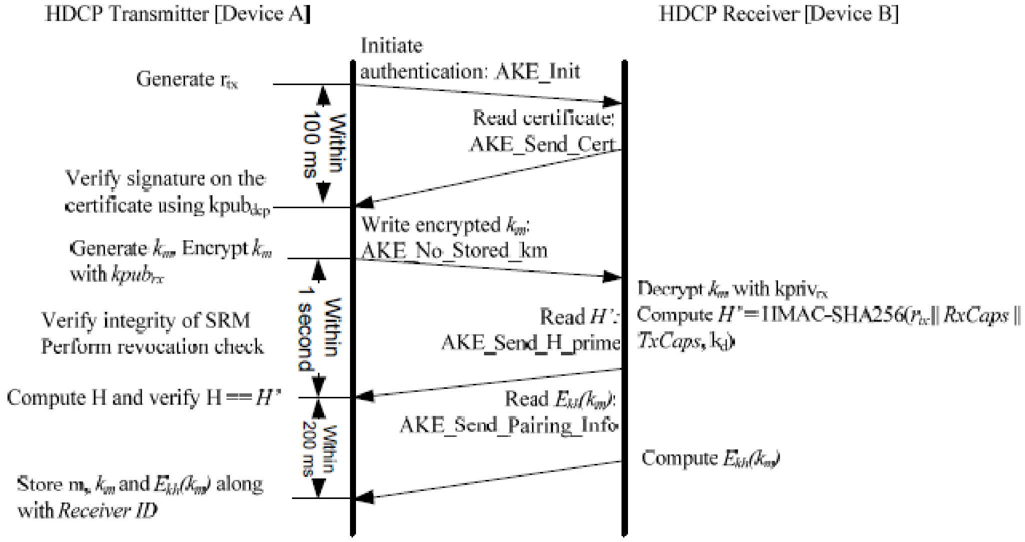
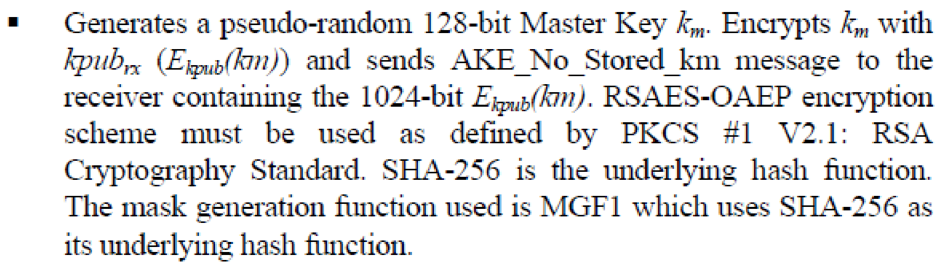
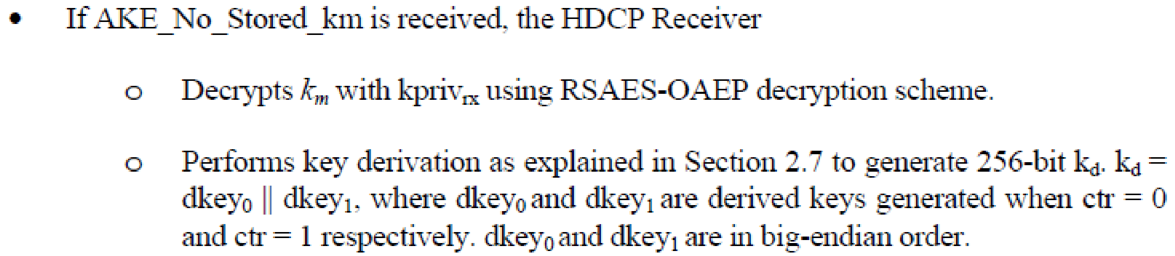


Figure 2.1. Authentication and Key Exchange (Without Stored k_m)

Id. at 12.

	 - Generates a pseudo-random 128-bit Master Key k_m. Encrypts k_m with $kpub_{rx}$ ($E_{kpub}(km)$) and sends AKE_No_Stored_km message to the receiver containing the 1024-bit $E_{kpub}(km)$. RSAES-OAEP encryption scheme must be used as defined by PKCS #1 V2.1: RSA Cryptography Standard. SHA-256 is the underlying hash function. The mask generation function used is MGF1 which uses SHA-256 as its underlying hash function. *Id.* at 13.  - If AKE_No_Stored_km is received, the HDCP Receiver ○ Decrypts k_m with kpriv$_{rx}$ using RSAES-OAEP decryption scheme. ○ Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = dkey$_0$ || dkey$_1$, where dkey$_0$ and dkey$_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. dkey$_0$ and dkey$_1$ are in big-endian order. *Id.* at 14. The Accused Product decrypts k_m using k_h when the transmitter (first device) previously stored a k_m corresponding to the Accused Product.

"create a second signal, wherein the second signal is derived from a secret known by the second device;"

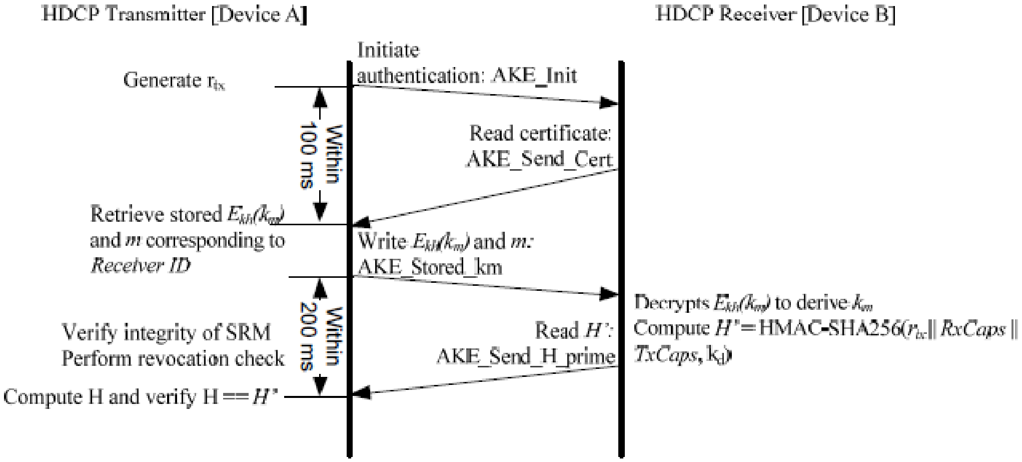


Figure 2.2. Authentication and Key Exchange (With Stored k_m)

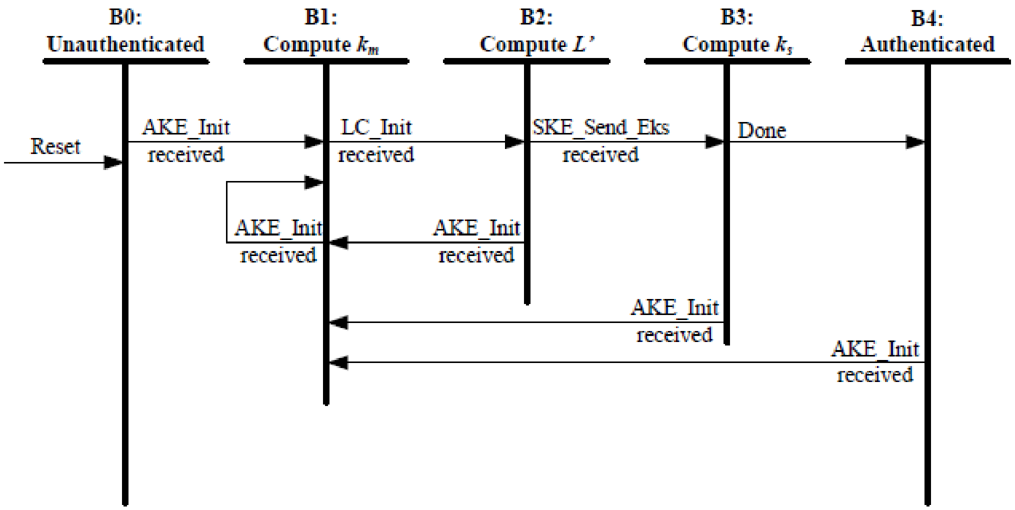
Id. at 12.

- Sends AKE_Stored_km message to the receiver with the 128-bit $E_{kh}(k_m)$ and the 128-bit m corresponding to the *Receiver ID* of the HDCP Receiver

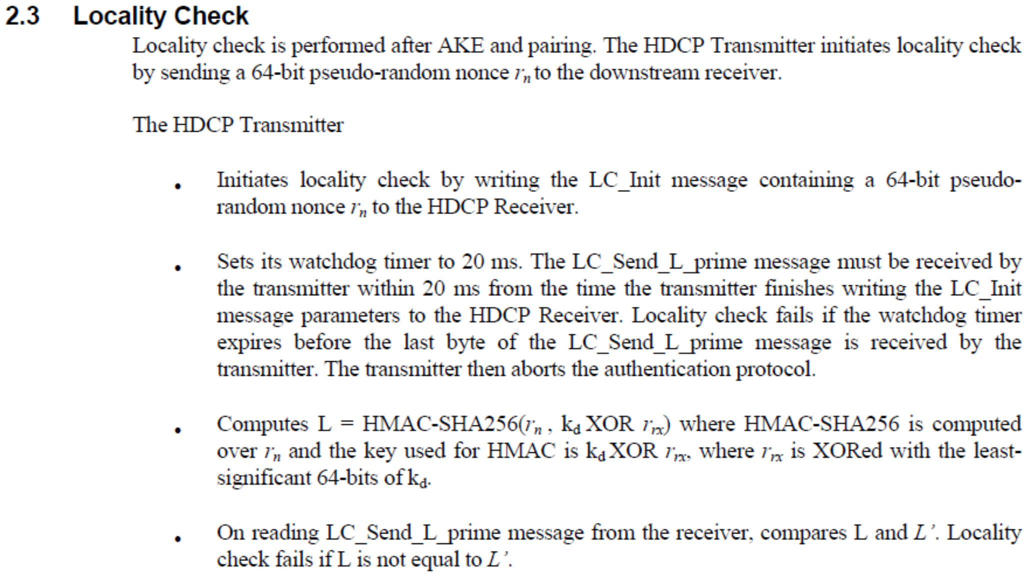
Id. at 14.

- If AKE_Stored_km is received, the HDCP Receiver
  - Computes 128-bit k_h = SHA-256($kpriv_{rx}$)[127:0]
  - Decrypts $E_{kh}(k_m)$ using AES with the received m as input and k_h as key in to the AES module as illustrated in Figure 2.3 to derive k_m.
  - Performs key derivation as explained in Section 2.7 to generate 256-bit k_d. k_d = $dkey_0$ || $dkey_1$, where $dkey_0$ and $dkey_1$ are derived keys generated when ctr = 0 and ctr = 1 respectively. $dkey_0$ and $dkey_1$ are in big-endian order.

Id. at 15.

	See also:  **Figure 2.13. HDCP Receiver Authentication Protocol State Diagram** *Id*. at 31. **State B2: Compute L'.** The HDCP Receiver computes L' required during locality check and makes the LC_Send_L_prime message available for reading by the transmitter. *Id*.

"provide the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device; and"

provide the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device; and	The instructions of the Accused Product are arranged to provide the second signal, *e.g.*, L', to the first device (transmitter) after receiving the first signal, *e.g.*, the LC_Init message including r_n. The Accused Product provides the second signal to the first device using, *e.g.*, the LC_Send_L_prime message, and the second signal is received by the first device.  **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. HDMI HDCP 2.2 at 16.

"provide the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device; and"

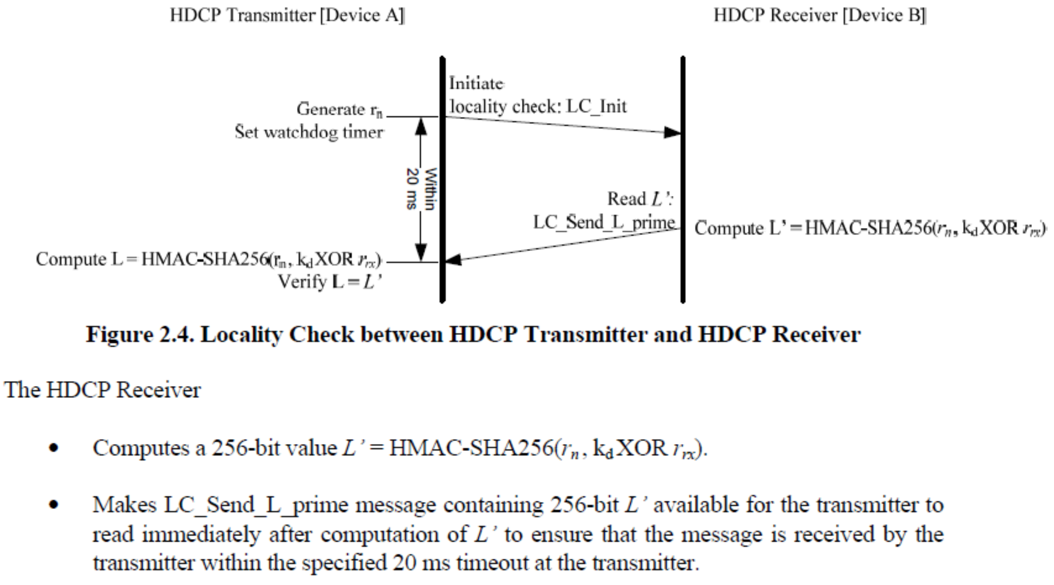
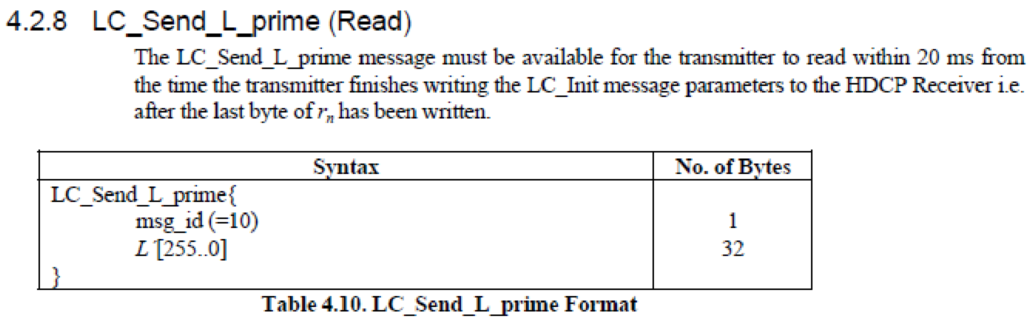


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

Id. at 17.



4.2.8 LC_Send_L_prime (Read)

The LC_Send_L_prime message must be available for the transmitter to read within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver i.e. after the last byte of r_n has been written.

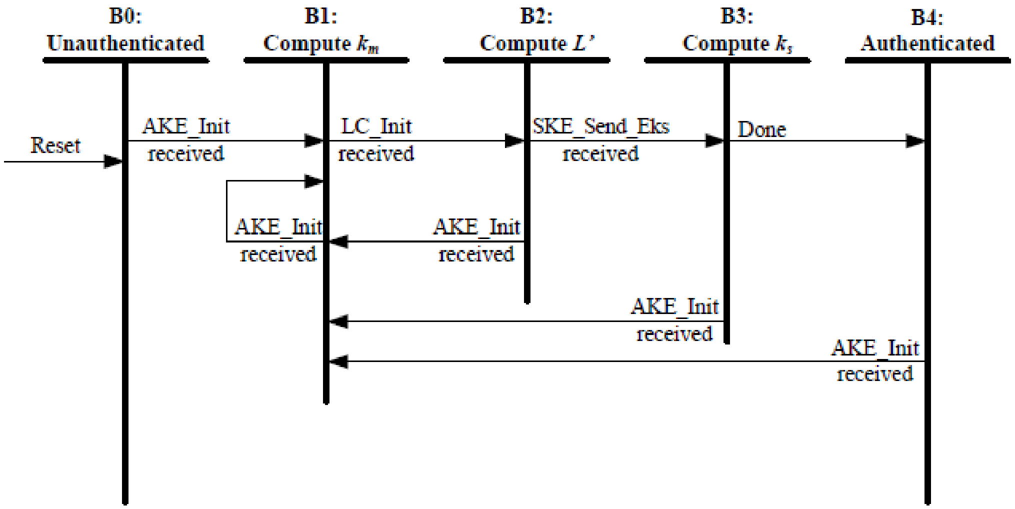
Syntax	No. of Bytes
LC_Send_L_prime{	
msg_id (=10)	1
L'[255..0]	32
}	

Table 4.10. LC_Send_L_prime Format

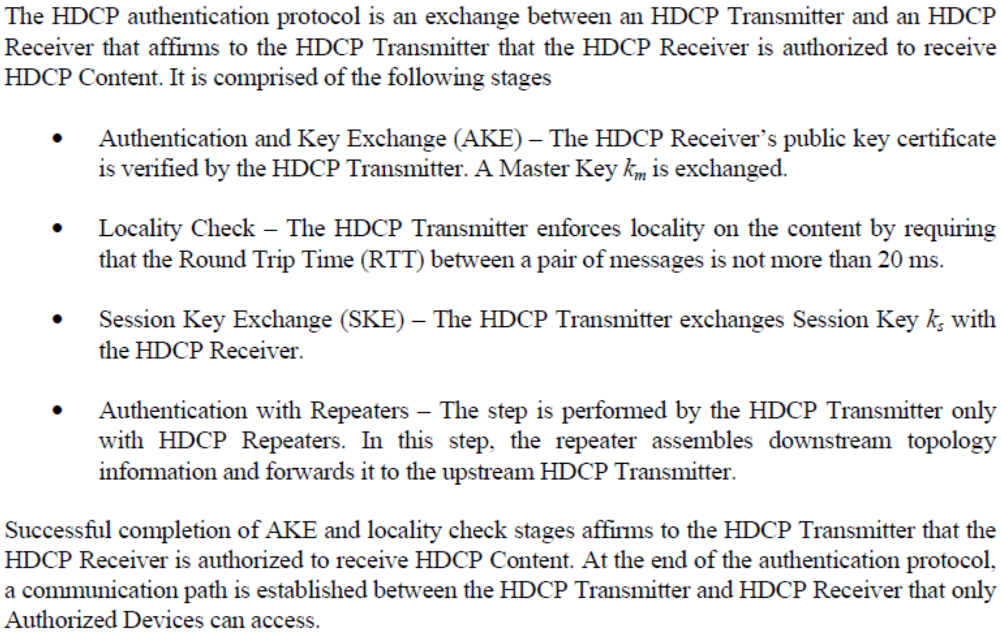
Id. at 59.

See also:

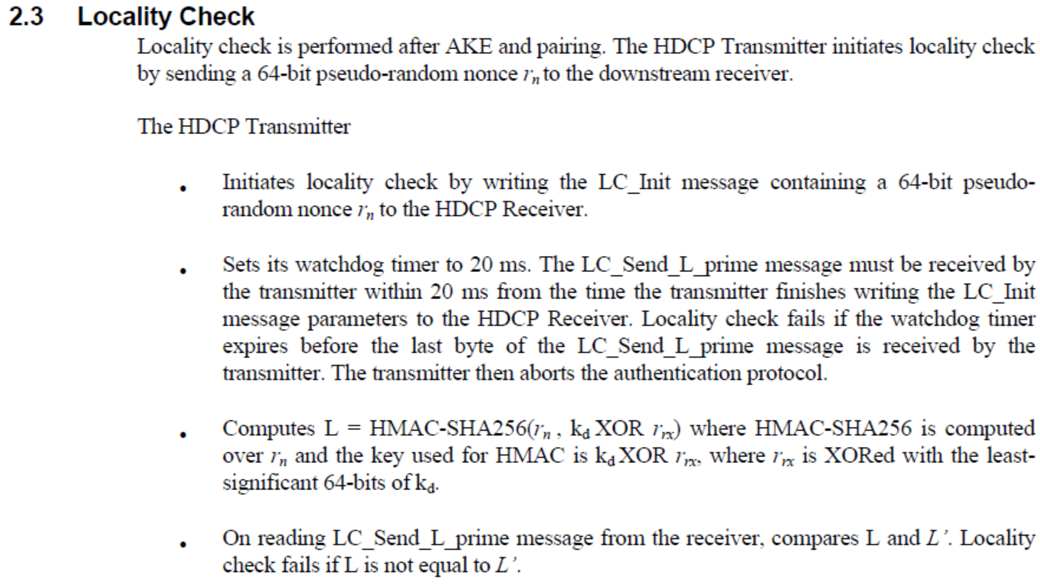
"provide the second signal to the first device after receiving the first signal, wherein the second signal is received by the first device; and"

	 **Figure 2.13. HDCP Receiver Authentication Protocol State Diagram** *Id*. at 31. > **State B2: Compute L'**. The HDCP Receiver computes L' required during locality check and makes the LC_Send_L_prime message available for reading by the transmitter. *Id*.

"receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time."

receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time.	The instructions of the Accused Product are arranged to receive the protected content from the first device when the first device determines that the second signal, *e.g.*, L', is derived from the secret and a time between the sending of the first signal, *e.g.*, the LC_Init message including r_n, and the receiving of the second signal is less than a predetermined time. The HDCP 2.2 Locality Check must be passed prior to session key exchange and establishment of a secure communications path for receipt of protected content by the Accused Product.  The HDCP authentication protocol is an exchange between an HDCP Transmitter and an HDCP Receiver that affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. It is comprised of the following stages • Authentication and Key Exchange (AKE) – The HDCP Receiver's public key certificate is verified by the HDCP Transmitter. A Master Key k_m is exchanged. • Locality Check – The HDCP Transmitter enforces locality on the content by requiring that the Round Trip Time (RTT) between a pair of messages is not more than 20 ms. • Session Key Exchange (SKE) – The HDCP Transmitter exchanges Session Key k_s with the HDCP Receiver. • Authentication with Repeaters – The step is performed by the HDCP Transmitter only with HDCP Repeaters. In this step, the repeater assembles downstream topology information and forwards it to the upstream HDCP Transmitter. Successful completion of AKE and locality check stages affirms to the HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. At the end of the authentication protocol, a communication path is established between the HDCP Transmitter and HDCP Receiver that only Authorized Devices can access. HDMI HDCP 2.2 at 11. The Accused Product receives protected content after the first device, as part of the Locality Check, determines that: the L' received via the LC_Send_L_prime message is derived from the secret (as determined by matching L' to value L which is derived from the secret (*e.g.*, L is computed based on k_d, which is based on $dkey_0$ and $dkey_1$, each of which is based on the Master Key, k_m)); and a time between the sending of the LC_Init message and receiving L' via the LC_Send_L_prime message is less than a predetermined time of 20 ms.

"receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time."

	 **2.3 Locality Check** Locality check is performed after AKE and pairing. The HDCP Transmitter initiates locality check by sending a 64-bit pseudo-random nonce r_n to the downstream receiver. The HDCP Transmitter • Initiates locality check by writing the LC_Init message containing a 64-bit pseudo-random nonce r_n to the HDCP Receiver. • Sets its watchdog timer to 20 ms. The LC_Send_L_prime message must be received by the transmitter within 20 ms from the time the transmitter finishes writing the LC_Init message parameters to the HDCP Receiver. Locality check fails if the watchdog timer expires before the last byte of the LC_Send_L_prime message is received by the transmitter. The transmitter then aborts the authentication protocol. • Computes L = HMAC-SHA256(r_n , k_d XOR r_{rx}) where HMAC-SHA256 is computed over r_n and the key used for HMAC is k_d XOR r_{rx}, where r_{rx} is XORed with the least-significant 64-bits of k_d. • On reading LC_Send_L_prime message from the receiver, compares L and L'. Locality check fails if L is not equal to L'. *Id*. at 16.

"receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time."

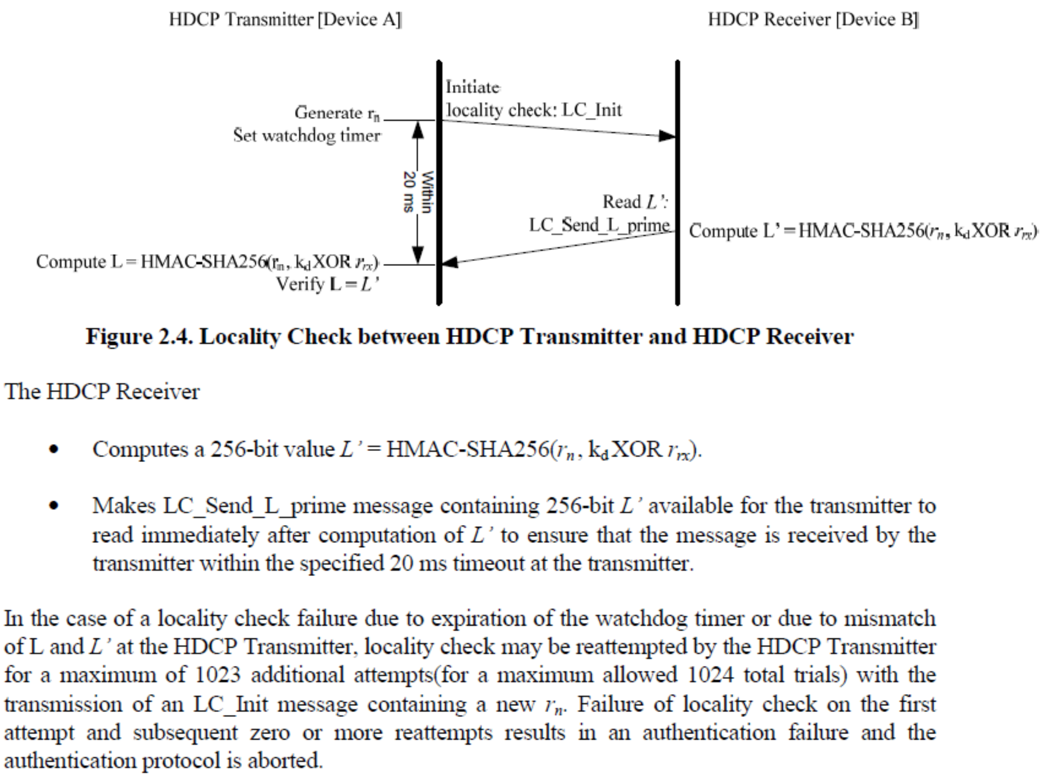


Figure 2.4. Locality Check between HDCP Transmitter and HDCP Receiver

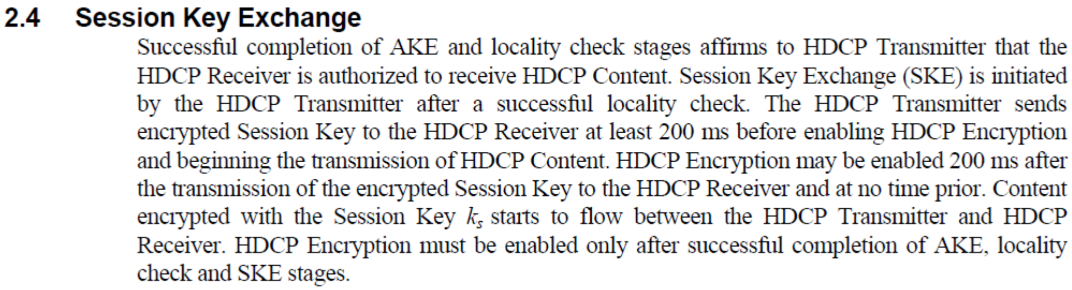
The HDCP Receiver

- Computes a 256-bit value $L' = \text{HMAC-SHA256}(r_n, k_d \text{ XOR } r_{rx})$.
- Makes LC_Send_L_prime message containing 256-bit L' available for the transmitter to read immediately after computation of L' to ensure that the message is received by the transmitter within the specified 20 ms timeout at the transmitter.

In the case of a locality check failure due to expiration of the watchdog timer or due to mismatch of L and L' at the HDCP Transmitter, locality check may be reattempted by the HDCP Transmitter for a maximum of 1023 additional attempts(for a maximum allowed 1024 total trials) with the transmission of an LC_Init message containing a new r_n. Failure of locality check on the first attempt and subsequent zero or more reattempts results in an authentication failure and the authentication protocol is aborted.

Id. at 17.

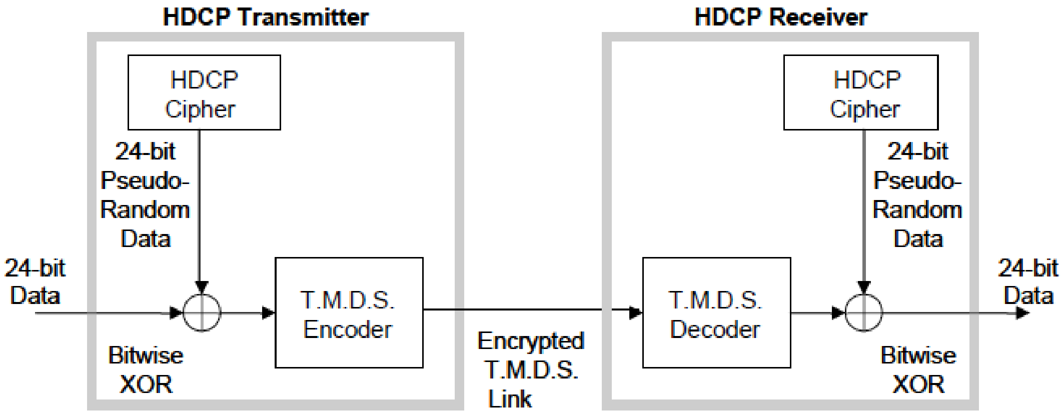
The Accused Product proceeds to session key exchange and receipt of the protected content after successful completion of the AKE stage and Locality Check.



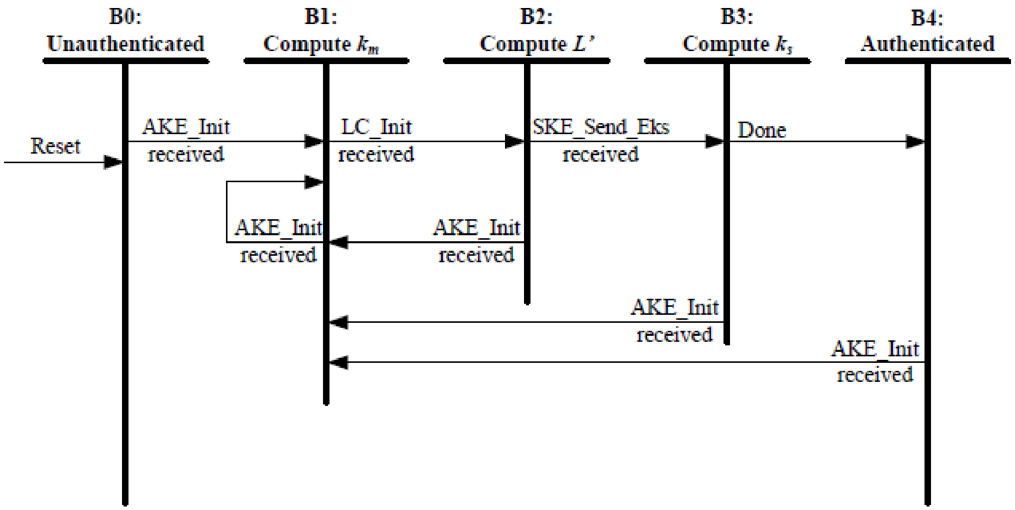
2.4 Session Key Exchange

Successful completion of AKE and locality check stages affirms to HDCP Transmitter that the HDCP Receiver is authorized to receive HDCP Content. Session Key Exchange (SKE) is initiated by the HDCP Transmitter after a successful locality check. The HDCP Transmitter sends encrypted Session Key to the HDCP Receiver at least 200 ms before enabling HDCP Encryption and beginning the transmission of HDCP Content. HDCP Encryption may be enabled 200 ms after the transmission of the encrypted Session Key to the HDCP Receiver and at no time prior. Content encrypted with the Session Key k_s starts to flow between the HDCP Transmitter and HDCP Receiver. HDCP Encryption must be enabled only after successful completion of AKE, locality check and SKE stages.

"receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time."

	Id. **3.1 Data Encryption** HDCP Encryption is applied at the input to the T.M.D.S. Encoder and decryption is applied at the output of the T.M.D.S. Decoder (Figure 3-1). HDCP Encryption consists of a bit-wise exclusive-or (XOR) of the HDCP Content with a pseudo-random data stream produced by the HDCP Cipher.  **Figure 3-1. HDCP Encryption and Decryption** *Id.* at 50. *See also*:

"receive the protected content from the first device when the first device determines that the second signal is derived from the secret and a time between the sending of the first signal and the receiving of the second signal is less than a predetermined time."

	 **Figure 2.13. HDCP Receiver Authentication Protocol State Diagram** *Id*. at 31. **State B3: Compute k_s.** The HDCP Receiver decrypts $E_{dkey}(k_s)$ to derive k_s. **Transition B3: B1**. Should the HDCP Transmitter write an AKE_Init while the HDCP Receiver is in State B3, the HDCP Receiver abandons intermediate results and restarts computation of k_m. **Transition B3: B4**. Successful computation of k_s transitions the receiver into the authenticated state. **State B4: Authenticated**. The HDCP Receiver has completed the authentication protocol. It must perform an ongoing link integrity check as described in Section 2.6. If the Receiver detects a synchronization mismatch between Transmitter and Receiver during the link integrity check, it must set the REAUTH_REQ bit in the *RxStatus* register. *Id*. at 32.