(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2009년01월21일
(11) 등록번호 10-0879606
(24) 등록일자 2009년01월13일

(51) Int. Cl.[9]
H04L 9/14 (2006.01) *G06F 17/00* (2006.01)
(21) 출원번호 10-2007-0020133
(22) 출원일자 2007년02월28일
심사청구일자 2007년02월28일
(65) 공개번호 10-2008-0079762
(43) 공개일자 2008년09월02일
(56) 선행기술조사문헌
논문 (2006.12)
(뒷면에 계속)
전체 청구항 수 : 총 3 항

(73) 특허권자
한남대학교 산학협력단
대전광역시 유성구 전민동 461-6
(72) 발명자
이동춘
대전 유성구 용계동 해송빌라 403호
박재표
서울 동작구 상도4동 211-468번지 102호
(74) 대리인
강홍구

심사관 : 이준석

(54) 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 방법

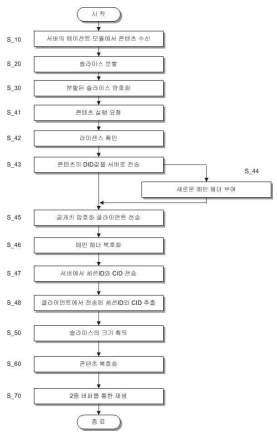
(57) 요 약

본 발명은 디지털 콘텐츠 제공 방법에 관한 것으로, 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 제공방법에 관한 것이다.

본 발명에 따르면 해쉬 체인 기법을 사용하여 여러 개의 비밀키를 사용하고, 전체 데이터를 암호화하는 기법과 세션값을 사용하여 사용자의 상호인증 프로토콜을 제공할 수 있으며, 디지털 콘텐츠 사용에 대한 인증을 부가하여 불법적인 사용자 비밀키 유출을 막을 수 있다.

또한 하나의 비밀키가 노출 되더라도 저작물 전체에 대한 복호화를 사전에 봉쇄함으로써 저작물을 재생할 수 없으며, 효율적인 버퍼 스케줄링을 수행하여 사용자에게 실시간 복호화 및 재생을 할 수 있도록 하고, 클라이언트에서 대용량의 동영상 데이터 파일 재생 시 복호화 시간을 포함한 재생 지연시간을 줄일 수 있는 디지털 콘텐츠 보안 인증 방법을 제공할 수 있다.

대 표 도 - 도3



(56) 선행기술조사문헌
JP2005217794 A
KR1020060048279 A
KR1020060066628 A
US20040243807 A1

특허청구의 범위

청구항 1

콘텐츠 메니저(111)를 포함하는 에이전트(Agent) 모듈(110), 암호화(Encryption) 프로세스(121)를 포함하는 시큐리티(Security) 모듈(120), 분석(Analysis) 프로세서(131)를 포함하는 분석 모듈(130), 디지털 컨텐츠를 저장하는 데이터베이스(140)로 구성된 서버(100);

상기 서버(100)에서 제공되는 디지털 콘텐츠를 복호화하는 프로세서(211)를 포함하는 시큐리티 에이전트(210), 상기 복호화된 디지털 콘텐츠를 재생하는 콘텐츠 플레이어(220)로 구성된 클라이언트(200)로 구성되어,

콘텐츠 제공자(CP)가 등록한 콘텐츠를 서버의 에이전트 모듈에서 수신 받는 단계(S_10);

상기에서 수신 받은 콘텐츠를 슬라이스로 분할하는 단계(S_20);

상기의 분할된 슬라이스를 암호화하는 단계(S_30);

클라이언트에서 콘텐츠 실행 요청 단계(S_41);

상기 콘텐츠의 LAU에서 라이선스를 확인하는 단계(S_42);

상기 콘텐츠의 DID 해쉬 값을 서버로 전송하는 단계(S_43);

상기 서버에서는 사용자의 복호화된 메인헤더를 통해서 상기 클라이언트의 DID값과 비교하여 일치하지 않으면 새로운 메인 헤더(MH)를 부여 받도록 하는 단계(S_44);

상기 클라이언트의 DID값과 동일하면 상기 서버에서는 상기 콘텐츠에 해당하는 DID값과 메인 헤더(MH)를 포함하여 공개키 암호화를 수행하고 상기 암호화된 공개키를 클라이언트로 전송하는 단계(S_45);

상기 클라이언트에서는 전송된 암호화된 공개키를 LAU의 라이선스로 메인 헤더(MH)를 복호화하는 단계(S_46);

상기 서버에서는 SSL를 통해 세션ID와 CID를 전송하는 단계(S_47);

상기 클라이언트에서는 전송된 세션ID와 CID를 추출하는 단계(S_48);

상기 클라이언트에서는 메인 헤더(MH)의 슬라이스 크기를 획득하는 단계(S_50);

상기의 CID를 이용하여 상기 콘텐츠를 복호화하는 단계(S_60);

상기 클라이언트에서는 상기 콘텐츠를 이중 버퍼를 통하여 재생하는 단계(S_70);

로 이루어지되,

상기 수신 받은 콘텐츠를 슬라이스로 분할하는 단계(S_20)는

콘텐츠의 재생 시간 및 화면 사이즈를 획득하는 단계(S_21);

상기 콘텐츠가 재생되는 동안 복호화할 수 있는 콘텐츠의 사이즈를 계산하는 단계(S_22);

상기 계산된 사이즈를 슬라이스로 분할하는 단계(S_23);

상기의 콘텐츠에 대해서 더 분할 할 수 있는 데이터가 있는지 확인하는 단계(S_24);

로 이루어짐을 특징으로 하는 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 방법.

청구항 2

청구항 제 1항에 있어서,

상기의 분할된 슬라이스를 암호화하는 단계(S_30)는

각각의 슬라이스를 슬라이스 헤더(SH)와 다수개의 블록으로 분할하는 단계(S_31);

상기의 각각의 블록에 대해서 0과 1로 매핑(mapping)하는 단계(S_32);

상기의 1로 매핑된 블록에 암호화하는 단계(S_33);

LAU와 콘텐츠 2차 ID(Foreign ID)를 포함하는 콘테이너 헤더(CH)를 포함하여 각각의 슬라이스를 결합하여 콘테이너를 생성하는 단계(S_34);

상기의 콘테이너에 메인 헤더(MH)를 결합하는 단계(S_35);

로 이루어짐을 특징으로 하는 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 방법.

청구항 3

청구항 제 1항에 있어서,

상기의 CID를 이용하여 상기 콘텐츠를 복호화하는 단계(S_60)는

상기의 콘테이너를 슬라이스 블록으로 분리하는 단계(S_61);

상기 분리된 블록을 상기에서 추출된 CID로 슬라이스 헤더(SH)를 복호화 하는 단계(S_62);

상기 슬라이스 헤더(SH)를 이용하여 각각의 블록 키를 생성하여 암호화된 블록을 복호화 하는 단계(S_63);

로 이루어짐을 특징으로 하는 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 방법.

명 세 서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

<19> 본 발명은 디지털 콘텐츠 제공 방법에 관한 것으로, 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 제공방법에 관한 것이다.

<20> 인터넷과 같은 컴퓨터 망과 컴퓨터 이용의 급격한 발달로 인하여 문서, 음성, 오디오, 영상, 동영상 등의 멀티미디어 데이터의 이용 및 보급이 일반화되고 있다.

<21> 그러나 이러한 멀티미디어 데이터들은 디지털이라는 속성으로 인하여 복사를 하게 되면 사실상 또 하나의 원본이 만들어지게 되므로 누구나 손쉽게 불법적인 복제를 통해서 이들 디지털 데이터를 획득할 수 있게 된다. 상기의 데이터 복제 방지를 위해서 인증이나 암호화와 같은 방법이 많이 활용되어 왔으나 해킹 기술이 향상되면서 이러한 기술을 무력화시키고, 한 사람이 적법하게 구입하여 배포하는 경우도 생기기 때문에 전자상거래 등 인터넷 기반의 사회경제 활동에 필요한 사용자 인증, 기밀성, 무결성, 저작권보호 등 인터넷 정보 유통환경의 안정성을 보장하기 위한 범국가적인 정보보호기반을 활성화해야 할 필요성이 높아졌다.

<22> 상기와 같은 디지털 콘텐츠 보호를 위한 방법으로 일반적인 인증방법을 도 1에 나타내었다.

<23> 도 1의 G는 메시지 M을 통신문 C로 변환하는 알고리즘이고, V는 수신한 통신문 C를 인증 하는 알고리즘이다. A와 B는 통신 시스템에 합법적으로 참여하는 사용자를 나타내며, X는 인터넷 통신로에 관여하여 메시지를 도청하거나 변경시키려는 비합법적인 사용자(eavesdropper)이다. 먼저 A는 알고리즘 G를 이용하여 메시지 M으로부터 C를 생성하여 인터넷을 통해서 B에게 전송한다. 이때 통신로를 거쳐 B에게 들어오는 통신문을 C라 한다면, 통신문 C를 수신한 B는 인증 알고리즘 V를 이용해서 복원한 메시지 M과 A가 송신한 메시지 M이 같은가, 또한 메시지를 보낸 측이 분명하게 A인지를 검사하여 그 결과를 출력한다.

<24> 상기와 같은 인증 방법은 일체형과 분리형으로 구분된다.

<25> 도 1a의 ①은 일체형 ②는 분리형을 나타낸 것으로, 일체형에서는 2개의 변환 알고리즘 P와 그 역변환 P^{-1}이 각각 키 KP와 K^{P-1}에 의해서 변환되는 암호화 복호화 알고리즘을 형성하고 있다. 출력 메시지 M이 의미가 있는 것이라면, 상대방 사용자 및 메시지가 정확한 것으로 인증될 수 있다.

<26> ②의 분리형은 메시지를 압축하는 것으로 hash function h를 사용하여 메시지를 압축한다. P와 P^{-1}은 일체형과

마찬가지로 키 K^{P}와 K^{P-1}에 의하여 변환되는 2개의 알고리즘이다. 또한 S는 인증자(authenticator)이며, 수신된 인증자 S를 복호화 하는 알고리즘 P^{-1}을 적용하여 얻어진 값 $P^{-1}(S)$과 수신된 메시지 M에 hash function h를 적용하여 얻은 값 h(M)가 일치 하는가 않는가에 의하여 메시지 및 상대 사용자를 인증할 수 있다.

<27> 공개키를 암호화 하는 방법으로, 공개키 암호화 알고리즘을 이용하는 방법, Wong 알고리즘을 이용한 방법 등이 있다.

<28> 공개키 암호 알고리즘을 이용한 방법은 이미지 전체를 암호화하지 않고서도 이미지에 대한 인증을 수행할 수 있는 방법을 제공한다. 인증으로 사용될 디지털 서명(digital signature) 등을 해쉬 함수를 이용하여 생성하고, 이를 비밀키를 사용하여 암호화한 다음 이미지와 함께 전송하는 방법이다.

<29> 이러한 공개키 암호 알고리즘을 이용한 방법에서 이미지에 대한 인증을 수행하는 과정을 자세히 살펴보면,

<30> 첫 번째, 인증에 사용될 디지털 서명(digital signature)을 생성한다. 이때의 디지털 서명은 암호학적 해쉬 함수를 사용하여 만들어진다.

<31> 서명 s = h(이미지)

<32> 두 번째, 비밀키(ka)를 사용하여 디지털 서명을 암호화한다.

<33> 인증자 = 공개키 암호(서명 s, 비밀키 ka)

<34> 세 번째, 암호화된 인증자와 이미지를 전송한다.

<35> 네 번째, 수신자는 공개키를 사용해서 암호화된 인증자를 복호화하여 송신자에게서 받은 이미지에 대한 해쉬 값을 구한다.

<36> 상기에서 구해진 원본 이미지의 해쉬 값과 수신된 해쉬 값을 비교하면 이미지가 변경되었는지를 확인할 수 있게 되고, 송신자의 공개키를 사용하여 복호화 했으므로 송신자가 누구인가를 알 수 있게 되어 인증이 가능하다.

<37> Wong 알고리즘을 이용한 방법은 이미지에 대한 인증을 수행하는 알고리즘으로, 이미지의 인증과 무결성을 제공하기 위하여 1998년 Ping Wah Wong이 제안한 것인데 해쉬 함수인 MD5를 이용한 방법과 공개키 암호 알고리즘을 이용한 방법이 있다.

<38> Wong은 암호학적 해쉬 함수인 MD5를 사용한 방법을 제안했는데, 이 방법은 정당하지 않은 키를 사용해서 워터마크를 확인하려하거나 이미지가 확대되거나 일부분만 잘려진 경우 등 원본 이미지의 변경이 일어났을 경우, 워터마크 대신 노이즈가 나타나게 함으로써 어느 위치에 수정이 일어났는지까지 확인 가능하다. 특히 이 방법은 암호학적인 해쉬 함수를 사용하므로 워터마킹 알고리즘의 안전성이 암호학적 해쉬 함수의 안전성에 의존하기 때문에 계산적으로 안전하다.

<39> Wong은 이러한 MD5를 이용한 워터마킹 알고리즘을 공개키 암호 알고리즘을 적용할 수 있도록 확장했다. 즉, 워터마크를 삽입할 때는 사용자의 비밀키(private key)를 사용하고, 워터마크를 추출하고자 할 때는 그 사람의 공개키(public key)를 사용하는 방법으로 삽입 과정을 살펴보면 다음과 같다.

<40> 첫 번째, $M_x \times N_x$ 크기의 이미지를 I x J (< 128) 픽셀 크기를 가지는 여러 블록들로 나눈다.

<41> 두 번째, 각 블록 X_r의 LSB 부분은 모두 제거한다.

<42> 세 번째, 이미지의 ID, 이미지의 크기, 블록의 인덱스(index), 블록 X_r의 남아있는 MSBs 부분을 해쉬 함수에 통과시켜서 디지털 서명을 생성한다.

<43> 네 번째, 생성된 서명과 삽입될 워터마크 이미지를 XOR 연산을 한다.

<44> 다섯 번째, 네 번째 단계에서 생성된 W_r을 사용자의 비밀키 K'으로 암호화한다.

<45> 여섯 번째, 다섯 번째 단계서 생성된 S_r을 이미지 블록 X_r의 LSB부분에 다시 삽입하여 워터마크가 삽입된 이미지 블록 XW_r을 생성한다.

<46> 그러나 상기의 방법들은 기존의 암호화 방법보다 다양한 키를 생성하는 알고리즘이 부재하고, 키 생성 알고리즘을 통해 각각 생성된 대칭키를 서버에 저장하지 않는 것으로 암호화의 수준이 저하되는 요인이 발생한다.

<47> 또한 용량이 큰 콘텐츠의 경우의 암/복호화 과정에 있어서 시간이 많이 소요되는 문제점이 발생한다.

발명이 이루고자 하는 기술적 과제

<48> 따라서 본 발명은 상기의 문제점을 해결하고자 안출된 것으로 저작권 보호기술의 대표적 분야인 디지털 워터 마킹(watermarking), DRM(Digital Rights Management : 저작권 관리 시스템), DOI(Digital Object Identifier : 디지털 객체 식별)의 기술을 중심으로 한 디지털 콘텐트의 보호관련 기술을 분석하고 시큐리티 에이전트와 해쉬 체인, 세션 키를 이용한 인증 방법을 제공하는 데 그 목적이 있다.

발명의 구성 및 작용

<49> 본 발명은 디지털 콘텐츠 제공 방법에 관한 것으로, 유/무선 통신망에서 시큐리티 에이전트와 해쉬 체인 및 세션 키 교환을 이용한 디지털 콘텐츠 보안 인증 제공방법에 관한 것이다.

<50> 본 발명에 따른 구성은 클라이언트/서버 구조로 구성되되, 서버는 에이전트 모듈; 암호화 모듈; 분석 모듈; 데이터베이스로 구성되고, 클라이언트는 복호화 처리기와 저작물 실행기로 구성된 시큐리티 에이전트로 구성된다.

<51> 상기 서버의 구성을 살펴본다.

<52> 도 2는 본 발명에 따른 시스템의 구성도를 나타낸 것이다.

<53> 도 2에 따르면, 콘텐츠 메니저(111)를 포함하는 에이전트(Agent) 모듈(110), 암호화(Encryption) 프로세스(121)를 포함하는 시큐리티(Security) 모듈(120), 분석(Analysis) 프로세서(131)를 포함하는 분석 모듈(130), 디지털 컨텐츠를 저장하는 데이터베이스(140)로 구성된 콘텐츠 제공 서버(100, Contents provider Server, 이하'서버'라함)

<54> 상기 서버(100)에서 제공되는 디지털 콘텐츠를 복호화(Decryption)하는 프로세서(211)를 포함하는 시큐리티 에이전트(210), 상기 복호화된 디지털 콘텐츠를 재생하는 콘텐츠 플레이어(220)로 구성된 클라이언트(200)로 구성된다.

<55> 상기 서버의 구성에서 상기 에이전트 모듈은 CP(Content Provider)에 등록된 컨텐츠를 서버의 암호화 모듈로 전송하고, 클라이언트의 시큐리티 모듈과 SSL(Secure Socket Layer)로 접속하여 세션값을 유지하면서, 클라이언트의 시큐리티 모듈에서 전송되는 모든 값들을 분석 모듈과 직접 통신하여 처리 및 관리하는 모듈이다.

<56> 상기 에어전트 모듈의 컨텐츠 관리자는 CP에서 전송된 컨텐츠의 ID(Contents ID, 이하 "CID"라 함)를 만들어서 컨텐츠 데이터베이스에 등록한다.

<57> 본 발명에 따른 디지털 콘텐츠의 제공 과정은 CP(Contents Provider)가 등록한 컨텐츠를 서버의 에이전트 모듈에서 수신하여 슬라이스 레이어로 상기 콘텐츠를 다수개로 나누고 상기 나눠진 다수개의 슬라이스를 암호화한다.

<58> 상기의 암호화된 콘텐츠는 사용자의 요구에 따라 제공하여 클라이언트에서 재생하게 되는데, 클라이언트에서 해쉬 값을 서버로 전송하면 사용자의 공개키로 암호화 하여 클라이언트로 전송하고, 상기 클라이언트에서는 상기 각각의 슬라이스 레이어 키를 생성하여 암호화된 블록을 복호화한다.

<59> 상기의 과정을 첨부된 흐름도에 따라 설명한다.

<60> 도 3은 본 발명에 따른 전체적인 흐름로이다.

<61> 첨부된 흐름도에 따라 설명하면,

<62> 콘텐츠 제공자(CP)가 등록한 콘텐츠를 서버의 에이전트 모듈에서 수신 받는 단계(S_10);

<63> 상기에서 수신 받은 콘텐츠를 슬라이스로 분할하는 단계(S_20);

<64> 상기의 분할된 슬라이스를 암호화하는 단계(S_30);

<65> 클라이언트에서 콘텐츠 실행 요청 단계(S_41);

<66> 상기 콘텐츠의 LAU(licence Aquisition URL)에서 라이선스를 확인하는 단계(S_42);

<67> 상기 콘텐츠의 DID 해쉬 값을 서버로 전송하는 단계(S_43);

<68> 상기 서버에서는 사용자의 복호화된 메인헤더를 통해서 상기 클라이언트의 DID값과 비교하여 일치하지 않으면 새로운 메인 헤더(MH)를 부여 받도록 하는 단계(S_44);

<69> 상기 클라이언트의 DID값과 동일하면 상기 서버에서는 상기 콘텐츠에 해당하는 DID값과 메인 헤더(MH)를 포함하여 공개키 암호화를 수행하고 상기 암호화된 공개키를 클라이언트로 전송하는 단계(S_45);

<70> 상기 클라이언트에서는 전송된 암호화된 공개키를 LAU의 라이선스로 메인 헤더(MH)를 복호화하는 단계(S_46);

<71> 상기 서버에서는 SSL(Secure Socket Layer)를 통해 세션ID와 CID를 전송하는 단계(S_47);

<72> 상기 클라이언트에서는 전송된 세션ID와 CID를 추출하는 단계(S_48);

<73> 상기 클라이언트에서는 메인 헤더(MH)의 슬라이스 크기를 획득하는 단계(S_50);

<74> 상기의 CID를 이용하여 상기 콘텐츠를 복호화하는 단계(S_60);

<75> 상기 클라이언트에서는 상기 콘텐츠를 이중 버퍼(Buffer)를 통하여 재생하는 단계(S_70);

<76> 로 이루어진다.

<77> 상기의 단계를 상세하게 설명한다.

<78> CP가 등록한 컨텐츠를 서버 에이전트 모듈에서 수신(S_10)되면, 서버의 시큐리티 에이전트는 전 처리 단계인 슬라이스로 나주어 주는 과정(S_20)을 수행한다. 슬라이스 분할은 서버에서 받은 해당 저작물에 대한 시간과 화면 사이즈를 획득한 후, 소정의 타임 간격(interval)에 해당되는 동영상 파일의 크기를 먼저 계산한 후, 상기의 타임 간격 부분이 재생된다. 바람직하게 상기 타임 간격은 약 10초 정도가 적당하다.

<79> 상기 슬라이스로 분할하는 과정을 흐름도로 살펴본다.

<80> 도 4는 슬라이스로 분할하는 과정의 흐름도이다.

<81> 콘텐츠의 재생 시간 및 화면 사이즈를 획득하는 단계(S_21);

<82> 상기 콘텐츠가 재생되는 동안 복호화할 수 있는 콘텐츠의 사이즈를 계산하는 단계(S_22);

<83> 상기 계산된 사이즈를 슬라이스로 분할하는 단계(S_23);

<84> 상기의 콘텐츠에 대해서 더 분할 할 수 있는 데이터가 있는지 확인하는 단계(S_24);

<85> 상기에서 더 분할 할 수 있는 데이터가 없을 때까지 상기의 콘텐츠가 재생되는 동안 복호화 할 수 있는 콘텐츠의 사이즈를 계산하고 슬라이스를 분할하는 과정을 반복한다.

<86> 더욱 바람직하게는 남아있는 데이터 분량이 전에 생성된 슬라이스보다 작으면 잔량에 대한 복호화의 시간을 계산하지 않도록 구성됨이 마땅할 것이다.

<87> 콘텐츠의 첫 번째 슬라이스는 타임 간격으로 지정되어 암호화와 복호화를 거치지 않고 바로 실행되는 구간으로 상기 첫 번째 슬라이스가 재생되는 동안 두 번째 슬라이스의 복호화 구간을 지정한다.

<88> 따라서 첫 번째 슬라이스의 타임 구간을 재생하는 동안 다음의 슬라이스는 구간은 타임 구간 동안의 복호화 할 수 있는 구간을 다음의 슬라이스 분할된다.

<89> 다음의 도 4a는 상기의 과정을 도면으로 나타낸 것으로 콘텐츠를 n개의 슬라이스로 분할하는 과정을 나타낸 것이다.

<90> 즉, 첫 번째 슬라이스(G1) 구간은 일정 구간으로 정해져 있으며, 다음의 슬라이스는 첫 번째 슬라이스(G1)가 재생되는 동안 클라이언트에서 복호화 할 수 있는 시간적 데이터 량이 되어 두 번째 슬라이스(G2)로 분할되고, 두 번째 슬라이스(G2)가 재생되는 동안 다음의 복호화 될 구간을 계산하여 세 번째 슬라이스(G3)로 분할된다. 상기와 같은 방법으로 해당 콘텐츠의 슬라이스를 n개로 나누어 저장을 하게 된다.

<91> 바람직하게는 상기 콘텐츠가 재생되는 동안에 다음의 슬라이스를 분할함에 있어서 다음의 복호화 전체 구간을 다음의 슬라이스 구간으로 정하는 것보다 일정한 비율로 지정하여 다음의 슬라이스 구간을 분할해야 한다. 이유는 복호화하는 과정에서 클라이언트의 시스템 사양, 네트워크의 상황 등에 따라 전(前) 구간의 슬라이스가 재생되는 동안 후(後) 슬라이스를 복호화 할 수 없는 경우가 발생될 수 있기 때문이다. 상기의 일정 비율은 당업자에 따라 변경될 수 있음은 당연하다.

<92> 슬라이스 분할 과정 이후의 과정은 암호화 과정이다.

<93> 상기에서 콘테츠가 n개의 슬라이스로 분할되고 각각의 슬라이스에 대해서 암호화를 하는 과정을 살펴본다.

<94> 상기의 암호화 과정에서 타임 간격(interval)에 해당되는 G1 슬라이스는 암호화를 하지 않고, 다음의 슬라이스(G2)부터 암호화를 적용한다.

<95> 콘텐츠 재생시 상기의 G1 슬라이스는 암호화가 되어 있지 않기 때문에 바로 재생을 할 수가 있으며, 상기 G1의 슬라이스가 재생되는 동안 G2의 슬라이스가 복호화 된다.

<96> 암호화 과정은 도 5의 나타내었으며, 첨부된 흐름도에 따라 설명한다.

<97> 각각의 슬라이스를 슬라이스 헤더(SH)와 다수개의 블록으로 분할하는 단계(S_31);

<98> 상기의 각각의 블록에 대해서 0과 1로 매핑(mapping)하는 단계(S_32);

<99> 상기의 1로 매핑된 블록에 암호화하는 단계(S_33);

<100> LAU와 콘텐츠 2차 ID(Foreign ID)를 포함하는 콘테이너 헤더(CH)를 포함하여 각각의 슬라이스를 결합하여 콘테이너를 생성하는 단계(S_34);

<101> 상기의 콘테이너에 메인 헤더(MH)를 결합하는 단계(S_35);

<102> 로 이루어진다.

<103> 상기의 과정을 도면으로 살펴보면 도 5a와 같다.

<104> 첨부된 도 5의 흐름도와 도 5a에 의해 부연설명하면 상기에서 암호화 조건은 연속적으로 암호화를 시키지 않는 블록은 없어야 하며, 전체 암호화된 블록은 50% 이상이 되어야 바람직하다. 그리고 암호화 할 슬라이스 블록은 1로 매핑(mapping)을 시키고, 암호화 하지 않을 슬라이스 레이어 부분을 0으로 매핑 시킨다.

<105> 상기의 슬라이스 헤더(SH)는 CID 값으로 다시 암호화 시켜 열어볼 수 없도록 만들어 둔다. 상기 슬라이스 헤더 다음의 블록 중에서 1로 매핑된 블록을 암호화 하는데, 암호화 키는 다음의 식으로 생성한다.

<106> $KEY = H (CID \,||\, S_b \,||\, n \,||\, EB)$ 식 1

<107> 헤더 정보(SH)와 CID로 해쉬한 값을 부분 슬라이스 블록의 1로 매핑된 부분만을 상기의 식 1의 키를 이용한 대칭키 암호 방법으로 암호화를 시키며, 헤더 정보 역시 CID 값으로 암호화를 시킨다.

<108> 암호화 과정을 거친 슬라이스 블록을 묶어서 컨테이너를 생성하고, 상기의 컨테이너 헤더(CH)는 LAU (License Acquisition URL)와 컨텐츠의 2차 ID (Foreign Content ID)로 구성이 되어 있으며, 상기 컨테이너는 웹사이트를 통해서 사용자가 다운 받을 수 있다.

<109> LAU에는 라이선스(License)를 획득할 수 있는 URL이 포함되어 있으며, 암호화된 컨텐츠를 재생시킬 때 해당 컨텐츠의 2차 ID의 값으로 해당 LAU의 라이선스가 있는지 확인을 하고, 만약 라이선스가 없다면 라이선스를 받을 수 있는 웹페이지 URL로 이동하기 위하여 넣어둔 값이며, 컨테이너 헤더는 암호화를 시키지 않는다.

<110> 상기의 메인 헤더(MH)를 별도로 구성을 해야 하는데, 상기의 메인 헤더(MH)는 클라이언트의 DID를 해쉬한 값을 저장할 공간과 각각의 슬라이스 G1~Gn의 시작 바이트를 기록해 놓은 파일이다.

<111> 상기와 같이 암호화된 콘텐츠를 클라이언트에서 복호화하여 재생하는 과정을 설명한다.

<112> 도 3의 단계 S_41 내지 단계 S_48은 복호화 전처리 과정을 나타낸 것이다.

<113> 동영상 컨테이너를 다운로드 받은 클라이언트에서 컨테이너를 실행(S_41)시키면 콘테이너 헤더(CH)에 있는 LAU를 통하여 라이선스(License)를 확인(S_42)한 후에 클라이언트의 DID의 해쉬 값을 전송(S_43)하면, 서버의 에이전트 모듈에서 상기 DID 해쉬 값을 해당 컨테이너의 MH에 포함하여 사용자의 공개키로 암호화 하여 클라이언트로 전송(S_45)해 준다. 그다음 클라이언트 에이전트는 사용자의 인증서를 바탕으로 컨텐츠의 메인 헤더파일 복호화 과정(S_46)을 수행한다.

<114> 상기 클라이언트에서는 자신의 개인키로 암호화된 헤더를 복호화 하더라도 슬라이스 레이어의 헤더와 각각의 슬라이스 레이어의 난수 블록은 해당 컨텐츠의 CID를 알 수가 없기 때문에 복호화 시킬 수가 없다.

<115> MH를 복호화 한 후, CID 값을 얻기 위해서는 서버에서 SSL을 통해 클라이언트 세션 ID값과 컨텐츠 헤더 파일을

XOR한 값, 즉 Temp ID(임시 ID)값을 전송 해 주게 되며, 클라이언트에서는 Temp ID값을 다시 Session ID와 XOR 시켜 CID값을 추출하게 된다.

<116> 사용자의 개인키로 복호화된 MH를 통해서 클라이언트의 DID값과 비교를 한 후, DID값이 일치하지 않는다면 복호화 작업을 중단하고, 새로운 MH를 부여(S_44) 받도록 한다.

<117> 상기에서 메인 헤더(MH)와 CID를 전부 획득하였다면, 상기의 메인 헤더(MH)에 있는 슬라이스 G1~Gn의 사이즈를 획득(S_50)한 후, 결합된 컨터이너를 슬라이스 블록으로 분리한 후, CID로 슬라이스 헤더 파일(SH)을 복호화 한 후, 슬라이스 헤더(SH)를 이용하여 각각의 슬라이스 블록 키를 생성하여 암호화된 블록을 복호화(S_60) 시킨다.

<118> 이에 복호화 과정을 상세히 살펴본다.

<119> 복호화 과정은 상기의 암호화 과정의 역 과정을 수행하여 이루어진다.

<120> 도 6은 본 발명에 따른 복호화 과정의 흐름도이다.

<121> 상기의 콘테이너를 슬라이스 블록으로 분리하는 단계(S_61);

<122> 상기 분리된 블록을 상기에서 추출된 CID로 슬라이스 헤더(SH)를 복호화 하는 단계(S_62);

<123> 상기 슬라이스 헤더(SH)를 이용하여 각각의 블록 키를 생성하여 암호화된 블록을 복호화 하는 단계(S_63);

<124> 로 이루어진다.

<125> 상기의 클라이언트에 위치한 시큐리티 에이전트는 복호화를 수행하기 위하여 암호화된 콘텐츠의 슬라이스 블록을 추출하고 비밀키로 복호화를 수행한 후, 버퍼 A와 버퍼 B에 번갈아 가며 저장하여 재생한다.

<126> 초기 재생 시간을 확보하기 위하여 첫 번째 슬라이스(G1)는 암호화 하지 않았기 때문에 상기 첫 번째 슬라이스(G1)는 바로 재생이 가능하며, G1을 재생할 때 두 번째 슬라이스(G2)의 복호화 과정이 동시에 수행이 된다.

<127> 상기의 방법으로 슬라이스 G2가 재생이 될 때 슬라이스 G3에 대한 복호화 과정이 반복되는 방법이다.

<128> 2중 버퍼를 이용한 재생 방법에 대해서 설명한다.

<129> 버퍼에는 전체 동영상이 플레이 되는 동안 지연되는 프레임을 계산하여 초기에 버퍼 사이즈를 결정한 후 플레이 하도록 하며, 2개의 버퍼를 사용하는 보상 2중버퍼 시스템을 사용한다.

<130> 원활한 재생을 위하여 초기에는 G1(타임 간격 값 : 약 10초 분량) 슬라이스를 재생하기 위해 버퍼 A에 저장하여 실행되어 지면, 슬라이스 G2 분량의 데이터를 복호화하여 버퍼 B에 저장한다. 버퍼 A에서 재생이 끝나면 시큐리트 에이전트는 버퍼 B의 데이터가 실행될 수 있도록 버퍼 B로 옮겨준다.

<131> 버퍼 A에서 버퍼 B로 바뀔 때 데이터의 끈김 현상이 발생하는데 G2, G3, G4의 첫 프레임이 임의의 수로 나누어진 완전치 않은 프레임이기 때문이다.

<132> 상기를 방지하기 위해서 G1, G2, G3의 마지막 프레임의 값을 버퍼 B에 붙여서 완전한 프레임으로 바꿔주기 위한 구성이 바람직하다.

<133> 또한 시큐리티 에이전트(210)에는 불법행위 감시기의 구성이 바람직하다. 상기의 불법행위 감시기는 사용자의 정보와 실행하고자 하는 동영상의 정보를 서버로 보내게 된다. 사용자의 불법적인 행위는 감시 인터페이스를 통해 서버의 데이터베이스에 저장된다. 인증된 사용자라 할지라도 사용권한에 따라 제한적인 사용을 위해 저작물 자체 암호화에 의해 저작물을 보호하게 된다.

<134> 사용자가 저작물에 대한 라이선스를 초과하여 사용하려고 시도하거나 사용자 임의의 복호화를 시도하는 등의 불법적인 사용 행위를 시도할 경우 이를 봉쇄하도록 저작물에 대한 지속적인 모니터링을 수행한다. 사용자가 저작물에 대해 불법적인 사용을 몇 회나 시도 했는지, 또한 사용권한 범위가 어느 정도인지 등 저작권 위배사례 수집 및 분석을 통하여 사용자에 대한 블랙리스트 관리와 각종 통계 정보를 계산하여 그 정보를 갱신 및 유지하는 작업을 수행한다.

<135> 이상 설명한 내용을 통해 당업자라면 본 발명의 기술 사상을 일탈하지 아니하는 범위에서 다양한 변경 및 수정이 가능함을 알 수 있을 것이다.

<136> 따라서, 본 발명의 기술적 범위는 실시예에 기재된 내용으로 한정하는 것이 아니라 특허청구의 범위에 의하여 정해져야 한다.

발명의 효과

<137> 종래의 DRM 시스템은 암호화 알고리즘으로 비밀키 암호 알고리즘을 사용하므로 암호화를 사전에 수행할 수 없고, 사용자가 디지털 콘텐츠를 다운로드할 때 암호화를 수행하여 다운로드 시간이 많이 소비되며, 사용자에 의하여 비밀키가 노출되면 저작권의 안전을 보장할 수 없었다.

<138> 따라서 본 발명은 상기의 문제점을 해결하기 위해 해쉬 체인 기법을 사용하여 여러 개의 비밀키를 사용하고, 전체 데이터를 암호화하는 기법과 세션값을 사용하여 사용자의 상호인증 프로토콜을 제공할 수 있으며, 디지털 콘텐트 사용에 대한 인증을 부가하여 불법적인 사용자 비밀키 유출을 막을 수 있다.

<139> 또한 하나의 비밀키가 노출 되더라도 저작물 전체에 대한 복호화를 사전에 봉쇄함으로써 저작물을 재생할 수 없으며, 효율적인 버퍼 스케줄링을 수행하여 사용자에게 실시간 복호화 및 재생을 할 수 있도록 하고, 클라이언트에서 대용량의 동영상 데이터 파일 재생 시 복호화 시간을 포함한 재생 지연시간을 줄일 수 있는 디지털 콘텐츠 보안 인증 방법을 제공할 수 있다.

도면의 간단한 설명

<1> 도 1은 디지털 콘텐츠 보호를 위한 방법으로 일반적인 인증방법.

<2> 도 1a는 일체형과 분리형의 일반적인 인증 방법.

<3> 도 2는 본 발명에 따른 시스템의 구성도.

<4> 도 3은 본 발명에 따른 전체적인 흐름로.

<5> 도 4는 슬라이스로 분할하는 과정의 흐름도.

<6> 도 4a는 콘텐츠를 n개의 슬라이스로 분할하는 과정.

<7> 도 5는 암호화 과정의 흐름도.

<8> 도 5a는 도 5의 흐름도에 대한 과정.

<9> 도 6은 본 발명에 따른 복호화 과정의 흐름도.

<10> ** 도면의 주요 부분에 대한 부호의 설명 **

<11> 100...서버

<12> 110...에이전트 모듈 111...콘텐츠 메니저

<13> 120...시큐리티 모듈 121...암호화 프로세서

<14> 130...분석 모듈 131...분석 메니저

<15> 140...데이터베이스

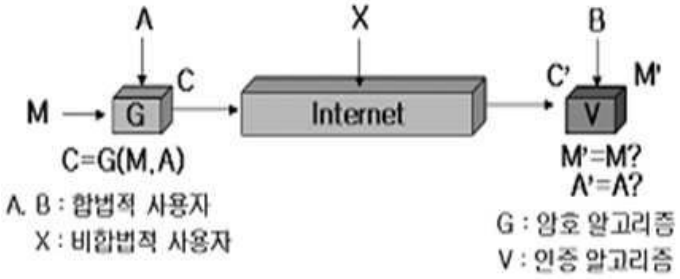
<16> 200...클라이언트

<17> 210...시큐리트 에이전트 211...복호화 프로세서

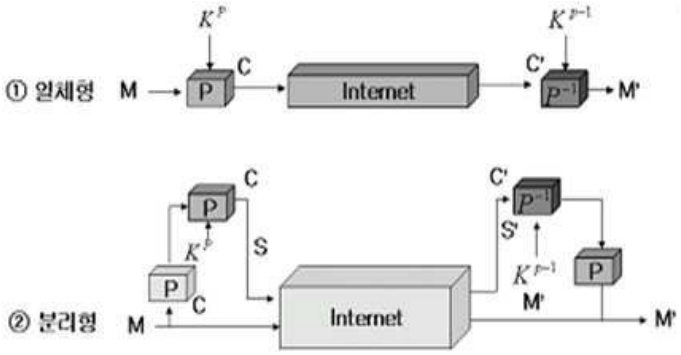
<18> 220...컨텐츠 플레이어

도면

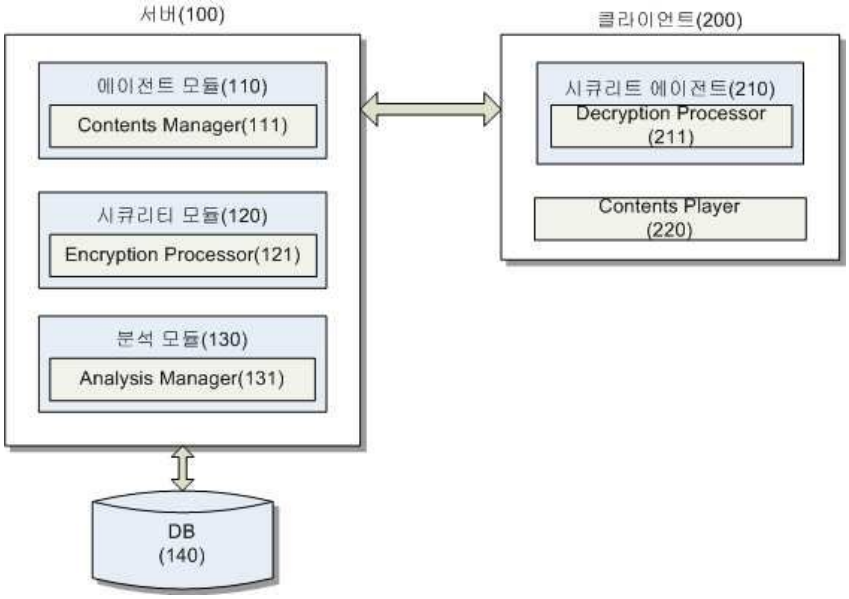
도면1



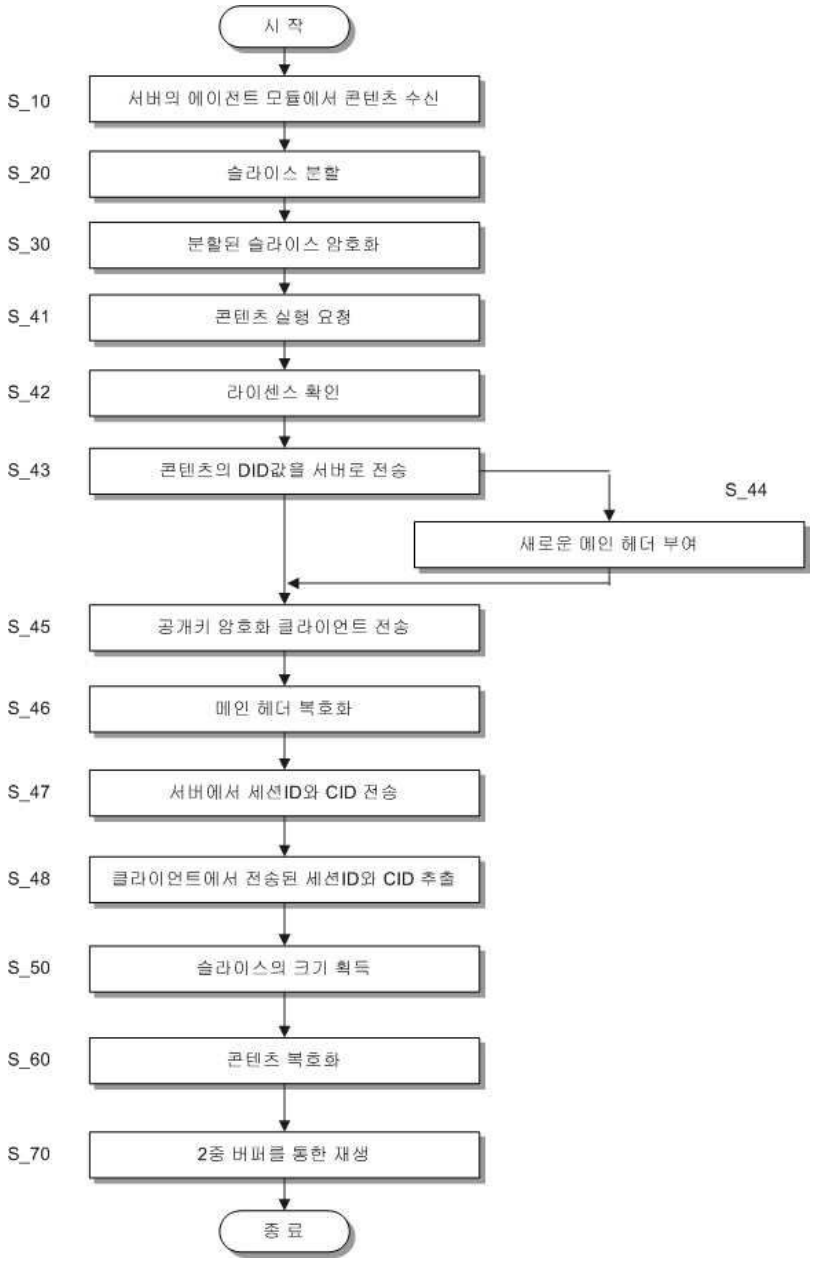
도면1a



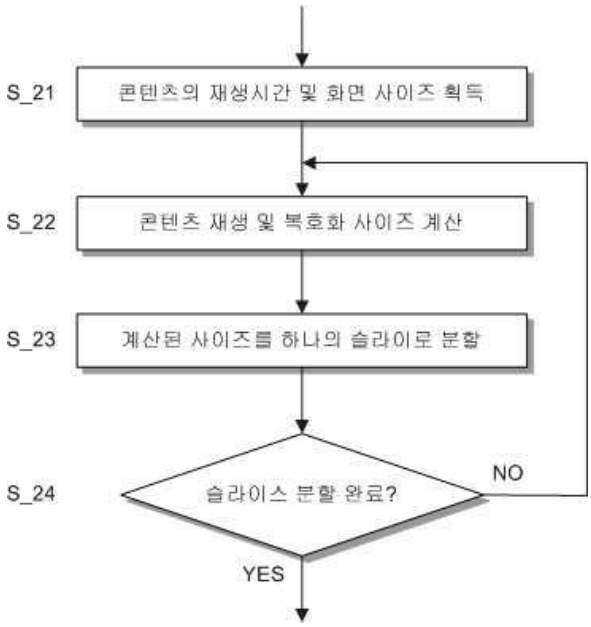
도면2



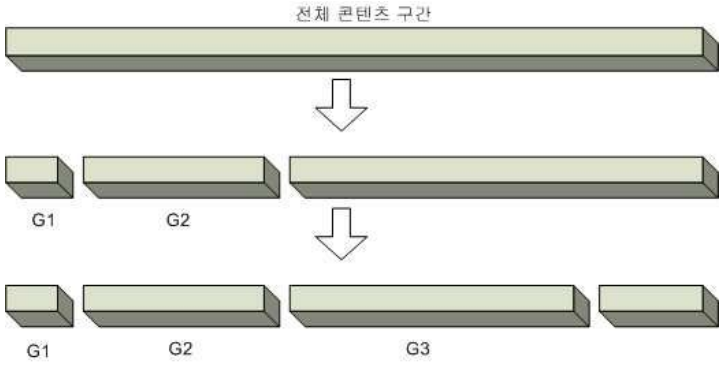
도면3


시 작
S_10
서버의 에이전트 모듈에서 콘텐츠 수신
S_20
슬라이스 분할
S_30
분할된 슬라이스 암호화
S_41
콘텐츠 실행 요청
S_42
라이센스 확인
S_43
콘텐츠의 DID값을 서버로 전송
S_44
새로운 메인 헤더 부여
S_45
공개키 암호화 클라이언트 전송
S_46
메인 헤더 복호화
S_47
서버에서 세션ID와 CID 전송
S_48
클라이언트에서 전송된 세션ID와 CID 추출
S_50
슬라이스의 크기 획득
S_60
콘텐츠 복호화
S_70
2중 버퍼를 통한 재생
종 료

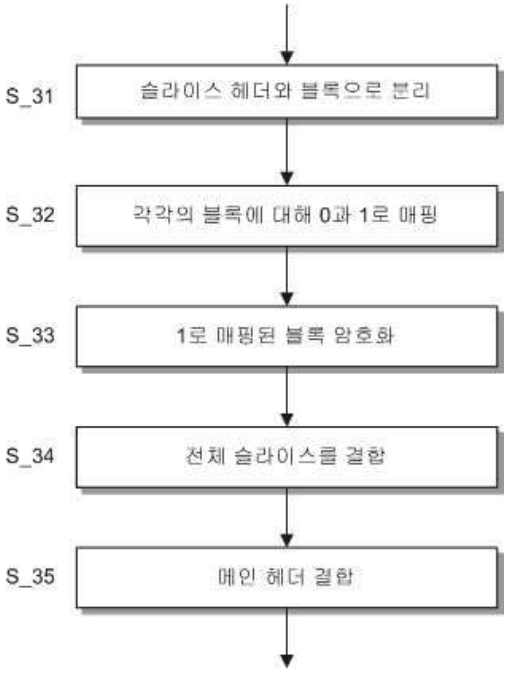
도면4


S_21
콘텐츠의 재생시간 및 화면 사이즈 획득
S_22
콘텐츠 재생 및 복호화 사이즈 계산
S_23
계산된 사이즈를 하나의 슬라이로 분할
S_24
슬라이스 분할 완료?
NO
YES

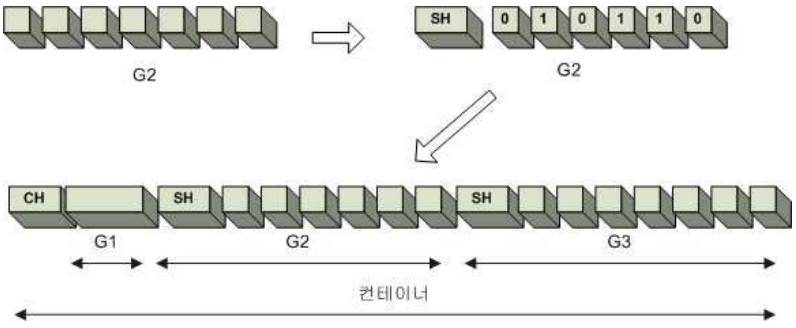
도면4a


전체 콘텐츠 구간
G1
G2
G1
G2
G3

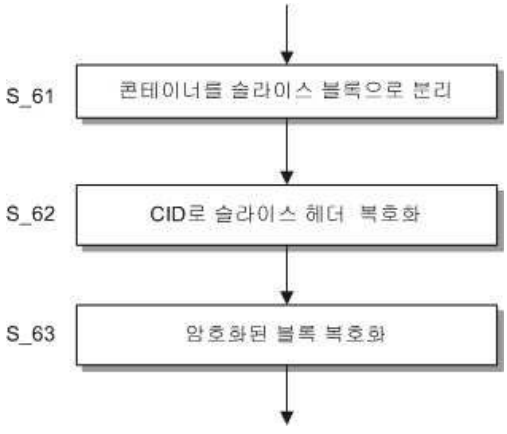
도면5


S_31
슬라이스 헤더와 블록으로 분리
S_32
각각의 블록에 대해 0과 1로 매핑
S_33
1로 매핑된 블록 암호화
S_34
전체 슬라이스를 결합
S_35
메인 헤더 결합

도면5a


G2
SH
0 1 0 1 1 0
G2
CH
SH
SH
G1
G2
G3
컨테이너

도면6


S_61
콘테이너를 슬라이스 블록으로 분리
S_62
CID로 슬라이스 헤더 복호화
S_63
암호화된 블록 복호화