(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2008-0066790
(43) 공개일자 2008년07월16일

(51) Int. Cl.
G06F 15/16 (2006.01) *G06F 17/40* (2006.01)
G06F 12/16 (2006.01) *G06F 21/00* (2006.01)
(21) 출원번호 10-2008-7011217
(22) 출원일자 2008년05월09일
심사청구일자 없음
번역문제출일자 2008년05월09일
(86) 국제출원번호 PCT/US2006/040389
국제출원일자 2006년10월12일
(87) 국제공개번호 WO 2007/044964
국제공개일자 2007년04월19일
(30) 우선권주장
60/725,812 2005년10월12일 미국(US)

(71) 출원인
데이터캐슬 코퍼레이션
미국 워싱턴 98065 스노퀄미 스위트 300 센터 비엘브이디 7829
(72) 발명자
섬너, 게리 스티븐
미국 워싱턴98045 노스 벤드 애비뉴 에스이 12815-452엔디
애먼스, 제이비, 마크
오스트레일리아 엔에스더블유 2450 사파이어 비치 코치맨스클로즈 42
리델, 마이크
오스트레일리아 비크 3121 리치몬드 번팅 스트리트 68
(74) 대리인
김해중, 윤석운, 홍순우

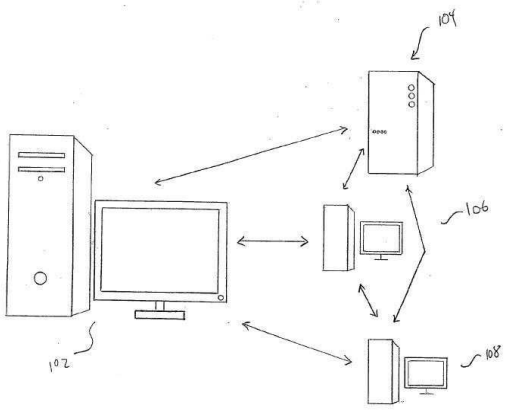
전체 청구항 수 : 총 10 항

(54) 데이터 백업 시스템 및 방법

(57) 요 약

본 발명의 실시예는 원격 데이터 백업 및 데이터 보존을 신뢰할 수 있고, 안전하고, 지리적으로 원격이고, 비용 효율적인 데이터 백업, 데이터 보존 및 백업되고 보존된 데이터 복원을 필요로 하는 개인, 소기업 또는 다른 조직에게 제공하는 웹서비스 기반 데이터 백업 및 데이터 보존 애플리케이션에 관한 것이다. 본 발명의 일 실시예에서, 개인 또는 소기업 클라이언트는 서비스 제공자와 데이터 백업 및 데이터 보존 서비스에 대해 계약한다. 서비스 제공자는, 그 결과, 안전하고, 신뢰할 수 있는 데이터 백업 및 데이터 보존을 개인 또는 소기업 클라이언트에게 제공하기 위해 원격 저장장치와 계약한다. 클라이언트측 애플리케이션은 클라이언트 컴퓨터에 다운로드되고, 클라이언트가 원격의 데이터 저장장치에서 국부적으로 암호화된 데이터를 저장하게 하도록 구성된다. 서비스 제공자뿐만 아니라 데이터 저장장치도 해독할 수 없고, 또는 클라이언트에 의해 저장된 정보에 접근할 수 없다. 또한, 암호키 또는 클라이언트에 의해 원격저장장치에 대한 데이터를 암호화하기 위해 사용되는 암호키는 클라이언트에 의한 서브시퀀트 복원을 위해 원격의 데이터 저장장치에 안전하게 저장되고, 클라이언트는 로컬 컴퓨터 시스템으로의 손상 또는 분실을 견뎌야만 한다. 그러나, 클라이언트 암호키는 이중으로 암호화된 형태로 저장되고, 서비스 제공자 또는 다른 데이터 저장장치에 의해 클라이언트의 암호키에 접근하는 것을 막는다. 본 발명의 일부 실시예는 또한 원격으로 저장된 임호화된 데이터에 대한 로컬 인덱싱 및 이미 원격으로 저장된 데이터로의 업데이트의 효과적인 저장을 제공한다.

대표도 - 도1



특허청구의 범위

청구항 1

서버측 부분; 및

클라이언트측 부분;을 포함하고,

서버측 부분은 백업 및 복원 요청을 수신하고,

복원 요청에 응답하여 암호화된 데이터 블록을 보내는 단계; 및 백업 요청에 응답하여 암호화된 데이터 블록 및 파일 서명을 저장하는 단계; 에 의해 백업 및 복원 요청을 처리하고,

클라이언트측 부분은

파일이 연속적인 백업을 지정하도록 하는 사용자 인터페이스를 제공하고,

연속적인 백업을 위해 지정된 파일의 변화, 파일 서명 계산, 파일 서명 비교에 의한, 백업 및 복원 동작을 위해 저장될 필요가 있는 블록 계산 및 백업 및 복원 동작에 대한 요청 발행을 검출하는 서비스 프로세스를 포함하고,

요청 및 데이터를 상기 서버측 부분과 교환하기 위한 전송 서비스 프로세스를 포함하는,

백업 및 복원 시스템.

청구항 2

제1항에 있어서,

파일 서명은 파일 서명 버전 및 블록 기술어의 순열을 포함하고, 각각의 블록 기술어는 블록 해시 및 블록 길이를 포함하는

백업 및 복원 시스템.

청구항 3

제1항에 있어서,

파일 서명은 파일로부터

자연 블록의 순열로 상기 파일을 분할하는 단계;

거의 고정된 크기의 블록의 순열로, 순서대로, 상기 자연 블록을 합체(coalesce)하는 단계;

각각의 거의 고정된 크기의 블록에 대해, 상기 블록의 길이 표시 및 상기 블록의 암호 해시를 포함하는 블록 기술어를 생성하는 단계; 및

상기 헤더를 생성된 상기 블록 기술어에 추가하는 단계;에 의해 생성되는

백업 및 복원 시스템.

청구항 4

제3항에 있어서,

각각의 거의 고정된 크기의 블록으로부터 완전히 지정된 거의 고정된 크기의 블록을 생성하기 위해 파일 암호키, 암호화 알고리즘 식별자 및 압축 알고리즘 식별자를 거의 고정된 크기의 블록에 추가하고, 상기 완전히 지정된 거의 고정된 크기의 블록에 상기 암호 해시 기능을 적용함으로써 각각의 암호 해시가 생성되는 백업 및 복원 시스템.

청구항 5

제1항에 있어서,

상기 서비스 프로세스는 이전 파일-변화 검출 이후에 변조된 파일을 검출하기 위해 파일의 현재 타임스탬프를

이전에 기록된 타임스탬프와 주기적으로 비교하는

백업 및 복원 시스템.

청구항 6

제1항에 있어서,

상기 서비스 프로세스는 상기 서버 부분으로 발송될 필요가 있는 블록을 결정하여,

파일에 대한 현재 파일 서명을 생성하는 단계;

파일이 마지막으로 백업된 뒤에 변경된 파일의 데이터 블록을 결정하기 위해 이전에 생성된 파일 서명에 현재 파일 서명을 비교하는 단계;

상기 서버측 부분에 의해 현재 저장되지 않은 상기 파일이 마지막으로 백업된 뒤에 변경된 상기 파일의 데이터 블록을 저장될 필요가 있는 데이터 블록으로 결정하는 단계; 및

상기 현재 파일 서명 및 저장될 필요가 있다고 결정된 데이터 블록을 상기 서버측 부분으로 전송하는 단계;에 의해 파일을 백업하는

백업 및 복원 시스템.

청구항 7

제1항에 있어서,

상기 서비스 프로세스는 블록이 상기 서버측 부분으로부터 요청될 필요가 있다고 결정하여,

복원될 파일의 인스턴스에 대한 파일 서명을 획득하는 단계; 및

클라이언트에 현재 사용할 수 없는 상기 파일 서명의 블록 해시에 의해 식별된 상기 블록을 결정하는 단계;에 의해 파일을 복원하는

백업 및 복원 시스템.

청구항 8

제1항에 있어서,

연속적인 백업을 위해 지정된 파일에서 데이터 블록이 몇 번이나 발생하는지와 상관없이, 계산된 블록 해시에 의해 식별된 각각의 데이터 블록은 서버측 부분에 의해서만 한번 저장되는

백업 및 복원 시스템.

청구항 9

제1항에 있어서,

상기 서버측 부분으로 전송된 각각의 데이터 블록은 상기 데이터 블록을 상기 서버측 부분으로 전송한 상기 클라이언트에게만 알려진 파일 암호키를 사용하여 암호화되어, 상기 서버측 부분이 상기 서버측 부분에 의해 저장된 클라이언트 데이터에 접근할 수 없는

백업 및 복원 시스템.

청구항 10

제1항에 있어서,

상기 서버측 부분은 클라이언트를 대신하여 이중으로 암호화된 파일 암호키를 저장하여, 상기 서버측 부분 또는 다른 클라이언트에게 상기 파일 암호키로의 접근을 제공하지 않고 분실한 파일 암호키를 복원할 수 있는 백업 및 복원 시스템.

명 세 서

기 술 분 야

<1> 본 발명은 데이터 백업 및 데이터 보관에 관한 것으로, 특히, 개별적이고 상업적인 컴퓨터 사용자가 데이터 파일을 포함하는 데이터를 웹 서비스 기반 애플리케이션을 통해 촉진하는 원격 데이터 저장장치에 백업 및 보관하도록 하는 웹 서비스 기반 데이터 백업 및 보관에 관한 것이다.

배 경 기 술

<2> 불과 30년 전에는, 수많은 대다수의 게임, 소기업, 중기업들이 워드 프로세싱 작업을 전자 타자기로 수행하였고 수기 및 타이핑된 종이의 개인 및 기업 관련 데이터를 색인된 폴더에 손으로 철하여 캐비닛 안에 저장하였다. 1970년대 후반 및 1980년대에는, 미니 컴퓨터 기반 워드 프로세싱 시스템 및 그 뒤에 개인용 컴퓨터가 널리 사용되게 되었으며, 전자 데이터 저장장치가 상대적으로 빠르게 보관용 캐비닛에 저장된 수기 및 타이핑된 서류를 대체하게 되었다. 그러나 많은 경우에, 전자 데이터는 플로피 디스크에 저장되는데, 즉, 데이터 백업 및 보존된 데이터를 저장하려는 그들의 유용성을 제한하는 초기의 매체 저장 장치 및 컴퓨터 시스템의 강력함이 부족하고 작은 용량 때문에 저장용 캐비닛과 같은 인클로저(enclosure) 안에 색인되고 저장된다.데이터 백업 및 보존된 데이터는 상대적으로 긴 시간 동안 안정적으로 저장되어야 한다. 종종, 백업되고 보존된 데이터가 다시는 필요하지 않을 수도 있지만, 그러나, 이러한 경우 백업 또는 보존된 데이터는 다음에 사용하기 위해 검색될 필요가 있고, 백업되거나 보존된 데이터를 검색하는 능력은 중요한 결과를 초래하고, 특정 경우에는, 기업 구조에 대해 치명적인 결과를 초래할 수 있다.

<3> 개인용 컴퓨터 및 기업 컴퓨터 시스템의 지속적인 개선과 함께 컴퓨터의 증가된 비용 효율에 따라, 데이터 백업 및 데이터 아카이브(archive)는 현재 가장 일반적으로 네트워크 컴퓨터 시스템에 의해 접근가능한 매체 저장 장치에 저장된다. 도1은 소기업 환경에서 데이터 백업 및 데이터 보존을 위한 옵션을 도시한다. 일반적으로, 고용인 또는 소기업 소유주는 대부분의 데이터 관련 업무를 고용인의 또는 기업 소유주의 개인 컴퓨터(102)에서 수행한다. 개인용 컴퓨터는 일반적으로 많은 데이터 백업을 허용하고 단일 컴퓨터 내의 완전한 디스크 미러링(mirroring) 및 많은 데이터 보존을 포함하는 다수의 디스크 드라이브와 함께 구매된다. 그러나, 소기업은 일반적으로 개인 컴퓨터의 네트워크 시스템 및 고용량이고 더욱 많이 활용될 수 있고 무결성 데이터 저장장치 서브시스템을 갖는 하나 이상의 서버(104)를 사용한다. 이러한 환경에서, 주로 PC 102를 사용하는 고용인 또는 기업 소유주는 네트워크를 통해 고용인 또는 기업 소유주 자신의 PC(102)내에 임의의 로컬 백업 및 보존에 더해 백업 데이터 저장 및 데이터 보존을 위해 다른 PC 106 및 108 또는 중앙집중 서버에 접근할 수 있다. 유사하게 가정의 사용자는 그들의 PC에 다수의 디스크 드라이브를 가질 수 있으며 둘 이상의 네트워크 연결된 컴퓨터를 통한 백업데이터 저장 데이터 보존을 허용하는 네트워크 연결된 다수의 PC 시스템을 가질 수도 있다. 또한, 데이터는 도1에 나타난 작은 시스템의 쓰기 가능한 CD 또는 DVD, 마그네틱 테이프, 또는 다른 유형의 물리적 저장 매체에 백업 및 보존되고, CD, DVD 또는 테이프는 먼 위치에 저장될 수 있다. 또한 이러한 실행은 정기적으로 처리되는 백업 및 보존, 원격 저장된 정보 관리 및 자주 잊히고 미루어지는 다른 수작업에 의존한다.

<4> 불행히도, 개인 및 기업 컴퓨팅의 현재 추세 및 발전은 도1에 나타난 불충분하고 위험한 소형 컴퓨터와 같은 소형 컴퓨터에서 데이터 백업 및 데이터 보전을 하려는 것이다. 애플리케이션 및 컴퓨터 시스템이 계속해서 더욱 커지고 더욱 유능해질수록, 정기적으로 생성되고 개인 또는 소기업 사용자에 의해 백업되고 보존될 필요가 있는 전자 데이터의 량은 급속히 증가한다. 또한, 컴퓨터 시스템의 증가하는 가격 수행 및 폭넓고 다양한 애플리케이션 프로그램의 증가하는 유용성의 결과로써 더욱 많은 기능 및 임무가 자동화되고, 더욱 많은 유형의 전자 데이터가 가정용 또는 소기업 컴퓨터 사용자에 의해 생성되며, 그중 많은 전자 데이터는 백업 및 보존될 필요가 있을 수도 있다. 새로운 규정 및 규칙은 소기업이 상대적으로 긴 시간 동안 확실하게 백업 데이터를 유지하기를 바란다. 예를 들어, 어떤 새로운 규칙이 의료 기록의 확실한 전자 기록장치를 필요로 하고, 다른 새로운 규칙이 비밀 상거래를 취급하는 회사의 다른 보안 관련 정보 및 이메일의 확실한 전자 저장장치를 필요로 한다. 이러한 규칙 및 규정은 막대한 추가 데이터 백업 및 데이터 보존 경비의 원인이 된다. 데이터 백업 및 데이터 보존은 지속적인 근면성 및 가정의 사용자와 소기업 부분에 대한 기술적 이해를 필요로 한다. 가정의 사용자 및 소기업은 대체로 백업되고 보존된 데이터는 소실되지 않고 또는 다양하고 상이한 이유로 회복할 수 없게 되지 않도록 보증하는 방식으로 데이터를 효과적으로 백업하고 보전하기 위한 기술적인 전문 기술, 시간 및 경계가 부족하다. 컴퓨터 납품업자, 오퍼레이팅 시스템 납품업자 및 다른 하드웨어, 소프트웨어 및 서비스 제공자에 의해 발전이 이루어지더라도 효과적이고 사용하기 쉬운 데이터 백업 및 데이터 보전은 서로 인터페이스하는 많은 상이한 구성요소를 필요로 할 수도 있으며, 많은 인터페이스는 시간에 대해 안정적이지도 않고 셋업 및 관리가 쉽지도 않다. 확실한 데이터 백업 및 데이터 보존은 단일 사이트에서 치명적인 데이터 손실을 막기 위해 둘 이

상의 지리적으로 먼 위치에 데이터가 저장될 것을 요구한다. 예를 들어, 소기업에서 데이터가 삼중 또는 사중의 중복 방식으로 백업되고 보존되더라도, 화재, 홍수 또는 지진은 모든 중복 저장된 데이터를 잃거나 회복할 수 없을 정도로 손상시킨다. 지리적으로 원격인 데이터 저장장치에 데이터를 백업하고 저장하는 것은 대개 가정의 사용자 및 소기업의 기술적 및 경제적 능력을 벗어난다. 마지막으로, 가정의 사용자 또는 기업이 믿을 수 있고 효과적인 데이터 백업 및 아카이빙 시스템을 생성하고 관리할 수 있더라도, 가정의 사용자 및 소기업이 우연 또는 고의의 승인되지 않은 액세스로부터 백업되고 보존된 데이터를 지키는 것은 대단히 어렵다. 이러한 데이터는 일반적으로 해커, 사업 경쟁자 및 사기 집단 및 조직에 의해 액세스된다. 모든 이러한 이유로, 가정의 사용자, 소기업 및 중간 규모의 기업 및 큰 조직들은 모두 사용하기 쉽고 믿을 수 있고, 비용효율이 높은 데이터 백업 및 데이터 저장장치에 대한 필요를 느끼게 되었다.

발명의 상세한 설명

<5> 본 발명의 실시예는 개별적인 개인, 소기업 및 믿을 수 있고, 안전하고, 지리적으로 원격이고, 비용효율이 높은 데이터 백업, 데이터 보존, 및 백업되고 보존된 데이터 검색을 필요로 하는 다른 조직에 원격의 데이터 백업 및 데이터 보존을 제공하는 웹서비스 기반 데이터 백업 및 데이터 보존 애플리케이션에 관한 것이다. 본 발명의 다른 실시예에서, 개인 또는 소기업 클라이언트는 데이터 백업 및 데이터 보존 서비스에 대해 서비스 제공자와 계약을 한다. 서비스 제공자는 그 다음에 안전하고 믿을 수 있는 데이터 백업 및 데이터 보존을 개인 또는 소기업 클라이언트에게 제공하기 위한 원격 데이터 저장장치와 계약을 한다. 클라이언트측 애플리케이션은 클라이언트 컴퓨터에 다운로드되고 클라이언트가 국부적으로 암호화된 데이터를 원격의 데이터 저장장치에 저장하게 한다. 서비스 제공자뿐만 아니라 데이터 저장장치는 해독할 수 없고 또는 그반대로 클라이언트에 의해 저장된 정보에 접근할 수 없다. 또한, 암호화 키 또는 원격 저장장치에 대해 데이터를 암호화하기 위해 클라이언트에 의해 사용되는 암호화 키는 원격의 데이터 저장장치에 안전하게 저장되고, 클라이언트에 의한 그 후의 복원을 위해 데이터 저장장치는 클라이언트가 로컬 컴퓨터 시스템에 손상 또는 손실을 견디도록 한다. 그러나, 클라이언트 암호화 키는 이중으로 암호화되는 방법으로 저장되어, 서비스 제공자 또는 데이터 저장장치 모두에 의해 클라이언트의 암호화 키에 접근하는 것을 막는다. 본 발명의 특정 실시예는 또한 원격으로 저장되고 암호화된 데이터 및 이미 원격으로 저장된 데이터 업데이트에 효과적인 저장장치에 대해 국부적인 인덱싱을 제공한다.

실 시 예

<33> 본 발명의 실시예는 웹서비스 기반 데이터 백업 및 데이터 보존 애플리케이션에 관한 것이다. 전술된 바와 같이, 전자 데이터 백업 및 데이터 보존이 가정의 사용자 또는 소기업에 일반적으로 사용되고 있지만, 믿을 수 있고, 안전하며 지리적으로 원격인 데이터 백업 및 데이터 보존에 대한 요구는 믿을 수 있는 데이터 백업 및 보존에 대한 요구 및 데이터 생성의 증가에 따라 계속 증가하고 있다.

<34> 전술된 바와 같이, 도1을 참고로 하여, 가정 또는 소기업 환경의 PC 사용자 모두는 가정의 원격 PC 뿐만 아니라 소기업 환경의 국부적 PC와 기업 환경의 서버의 하드웨어 및 소프트웨어에 접근할 수도 있다. 그러나, 도1을 참고로 하여 논의된 바와 같이, 네트워크된 가정용 컴퓨터 시스템 및 소기업 시스템은 데이터 백업 및 보존 요구에 대해 부적절하다.

<35> 도2에 나타난 바와 같이, 로컬 하드웨어 및 소프트웨어 리소스에 접근하고 하드웨어 및 소프트웨어 리소스가 근거리 통신망(local area network)에 대해 사용가능하다는 것에 덧붙여, PC 사용자는 인터넷 서비스 제공자에 의해 제공된 인터넷 접근 및 로컬 PC에서 실행되는 웹 브라우저 204를 통해 전세계로부터의 막대한 양의 HTML 인코드 정보 및 인터넷 기반 서비스에 접근할 수도 있다. 불행하게도, PC 사용자는 인터넷을 통해 천만 페이지의 정보에 접근할 수 있고, 다양한 상품 및 서비스를 구매하고 수신하기 위해 인터넷상의 기업과 기업의 거래 및 소매거래를 처리할 수도 있지만, 파일 이름 및 다른 파일 속성을 포함하는 잠재적으로 비밀인 정보를 원격 데이터 저장장치, 해커 및 인터넷을 통한 정보 전송을 가로채려는 사람들에게 노출되지 않고 사용자가 지속적으로, 안전하고, 투명하게 데이터 파일 및 데이터 저장장치를 조정하기 위한 다른 파일을 인터넷을 통해 백업 및 보존을 목적으로 업로드하는 것을 허용하는 알려진 방법이 현재는 없다.

<36> 애플리케이션과 애플리케이션 사이의 인터넷을 통한 거래를 위한 새로운 표준이 현재 개발중에 있다. 최근에 만들어진 표준의 모음은 "웹 서비스"라고 불린다. 웹 서비스는 현재 포트 80 및 443을 포함하는 특정 포트에 관련된 거래 또는 특정 유형의 동작을 규정하는 HTTP 기반 또는 HTTPS 기반, 및 XML 기반 프로토콜 의 모음으로 생각될 수 있다. 예를 들어, 웹 서비스 프로토콜은 특정의 한정된 업무를 실행하기 위해 서버 기반 컴퓨터와 상호작용하여 클라이언트 컴퓨터에서 실행되는 특정 애플리케이션 프로그램을 애플리케이션 프로그램에 허용하도록

규정될 수도 있다. 웹 서비스 기반 애플리케이션은 의료 정보를 암호화하고 전송하기 위한 클라이언트측 및 서버측으로 짝지어진 애플리케이션 프로그램을 포함할 수도 있다. 다른 웹 서비스 기반 애플리케이션은 동시에 발생하는 오디오 및 시각적인 정보가 두 개의 동등한 PC 사이에 전송되고 인터넷을 통해 상호연결된 두 개의 PC와 인터페이스하는 사용자 그룹 또는 사용자 사이의 비디오 회의를 허용하도록 방송 및 디스플레이되게 할 수도 있다. 컴퓨터 시스템의 가정의 사용자 및 소기업 사용자 모두로의 웹 브라우저 및 인터넷 접근의 유용성 및 웹 서비스 기반 애플리케이션의 급변은 데이터 저장장치를 조정하기 위한 믿을 수 있고 안전하고 비용효율이 높은 데이터 백업 및 데이터 보존 서비스와 함께 본 발명의 다수의 다양한 실시예를 자극한다.이러한 실시예는 웹 서비스 기반 데이터 백업 및 데이터 보존 서비스에 관한 것으로, 개인, 소기업 고용인 또는 다른 PC 사용자 또는 소형 컴퓨터 시스템이 백업 또는 보존용 데이터를 쉽고 비용효율이 높게 원격 데이터 저장장치에 인터넷상에서 웹 서비스 기반 애플리케이션을 거쳐 전송하고 필요할 경우 원격 데이터 저장장치로부터 백업되고 보존된 데이터를 검색하는 것을 허용한다.

<37> 도4는 도1의 종래의 예를 사용하여, 가정의 사용자, 소기업사용자 또는 다른 PC 또는 다른 소형 컴퓨터 시스템 사용자에게 사용할 수 있는 백업 및 보존 리소스를 예시한다. 도1을 참고로 하여 설명한 바와 같이, 데이터가 생성되고 생성된 데이터가 믿을 수 있게 백업 및 보존될 필요가 있는 가정 환경, 소기업 환경, 검색 환경 또는 다른 환경 내의 PC(102) 또는 다른 소형 컴퓨터 시스템의 사용자는 국부적인 PC(102) 내에 중복되게 데이터를 백업 및 보존하기 위한 국부적인 매체 저장장치, 다른 하드웨어 및 소프트웨어를 이용할 수도 있으며, 원격의 네트워크되는 PC들(106, 108)에 데이터를 백업 및 보존할 수도 있으며, 집중화된 서버 또는 다른 대규모 컴퓨터 소스(104)에 데이터를 백업 및 보존할 수도 있고, 본 발명에 따라, 데이터 백업 및 데이터 본존 웹 서비스를 지원하는 원격의 데이터 저장장치에 데이터를 백업 및 보존하기 위해 데이터 백업 및 데이터 보존을 하는 웹서비스 기반 서비스(402)를 이용할 수도 있다. 웹 서비스는 국부적인 PC(102)에서 실행되는 데이터 백업 및 데이터 보존 클라이언트측 애플리케이션 프로그램을 통해 로컬 PC(102)로부터 직접 액세스되거나, 집중화된 컴퓨팅 리소스(104) 또는 원격의 PC들(106, 108)을 통해 데이터 백업 및 데이터 보존을 하는 웹 서비스에 간접적으로 액세스할 수도 있다.

<38> 임의의 웹 서비스와 같은 데이터 백업 및 데이터 보존 웹 서비스는 한정된 웹 서비스를 함께 구성하는 동작, 원격의 함수 호출, 또는 다른 이와 같은 기능적인 인터페이스의 컬렉션(collection)으로써 간주될 수 있다. 본 발명의 다양한 실시예에서, 원격의 데이터 저장장치는 클라이언트측 데이터 볼트(vault) 애플리케이션을 실행하는 클라이언트 컴퓨터에 제1 웹 서비스 인터페이스를 제공하고, 원격의 데이터 저장장치에 의해 제공된 데이터 백업 및 데이터 보존 서비스와 클라이언트가 계약하여 파트너 서비스 제공자에게 제2 웹 서비스 인터페이스를 제공하는 데이터 볼트를 구현한다. 도4는 원격의 데이터 저장장치에서 실행되는 데이터 볼트 웹 서비스 기반 애플리케이션의 일 실시예에 의해 제공된 두 개의 웹 서비스 인터페이스를 예시한다. 데이터 볼트 애플리케이션은 제1 웹 서비스 인터페이스를 개별 프로토콜을 포함하는 클라이언트 컴퓨터에 제공하며, 개별 프로토콜은 클라이언트가 원격의 데이터 저장장치(402)에 클라이언트에 의해 저장된 파일 리스트를 검색하는 것을 허용하고, 원격의 데이터 저장장치(404)에 저장된 파일의 검색에 대한 준비를 요청하는 것을 허용하고, 데이터 저장장치(406)로부터 검색을 위해 요청된 파일을 실제로 검색하는 것을 허용하고, 원격 데이터 저장장치(408)에 저장을 위해 파일의 업로드에 대한 준비를 요청하는 것을 허용하고, 저장을 위해 원격 데이터 저장장치(410)에 파일을 실제로 업로드하는 것을 허용한다. 일실시예에서, 데이터 볼트 웹 기반 애플리케이션은 파트너 인터페이스를 써드파티(third party), 파트너 서비스 제공자에게 제공하며, 파트너 서비스 제공자가 데이터 볼트 애플리케이션(412)으로부터 장치 사용 정보를 획득하는 것을 허용하고, 파트너 서비스 제공자(414)를 통해 클라이언트에 대해 데이터 볼트 애플리케이션에 의해 구성된 장치를 리스트하는 것을 허용하고, 파트너 서비스 제공자(416)를 통해 클라이언트 컴퓨터를 위해 구성된 장치를 무력화하는 것을 허용하고, 파트너 서비스 제공자(418)의 클라이언트를 위해 구성된 장치를 작동시키는 것을 허용하고, 파트너 서비스 제공자(420)를 통해 클라이언트를 위해 구성된 장치를 제거하는 것을 허용하고, 파트너 서비스 제공자(422)의 클라이언트를 위해 새로운 장치를 생성하는 것을 허용한다. 다른 실시예에서, 추가적인 기능성이 제1 및 제2 데이터 볼트 웹 기반 인터페이스에 의해 제공될 수도 있고, 또 다른 실시예에서, 상이한 프로토콜 및 관련 동작, 원격의 함수 호출(procedure call), 또는 다른 기능 인터페이스의 컬렉션이 제공될 수도 있다. 일부 실시예에서, 웹 서비스 기반 데이터 백업 및 데이터 보존 서비스가 파트너 서비스 제공자가 필요없이 원격 데이터 저장장치에 의해 클라이언트에게 직접 제공될 수도 있다.

<39> 도5는 본 발명의 다양한 실시예를 지원하는 하나의 가능한 하이 레벨 하드웨어 구성을 예시한다. 도5에서, 클라이언트 컴퓨터(502)는 인터넷 기반 통신(508)에 대한 지원을 포함하는 운영체제(506)의 상부에 클라이언트 측 데이터 백업 및 데이터 보존 애플리케이션(504)을 동작시킨다. 운영체제(506)는 TCP/IP 프로토콜(512)의 상부에

HTTPS 510 프로토콜을 지원한 다음, 장치 드라이버(516)로 내부 버스를 통해 데이터를 전송하는 하나 이상의 장치 드라이버 고유 프로토콜(514) 위에 쌓이고, 결국, 전자 메시지를 원격 컴퓨터 지원 HTTPS 및 TCP/IP 프로토콜에 전송하고, 원격 컴퓨터 지원 HTTPS 및 TCP/IP 프로토콜로부터 전자 메시지를 수신한다. 파트너 데이터 백업 및 데이터 보존 애플리케이션(520)은 파트너 서비스 제공자의 컴퓨터(522)에서 동작한다. 전술된 바와 같이, 본 발명의 일 실시예에서, 클라이언트는 데이터 백업 및 데이터 보존 서비스를 위해 파트너 서비스 제공자와 계약을 한다. 일단 서비스가 구축되면, 클라이언트는 데이터를 저장 및 검색하기 위해 원격 데이터 저장장치(524)와 직접 통신한다. 일 실시예에서, 원격 데이터 저장장치(524)는 둘 이상의 지리적으로 분리된 컴퓨터 시스템(526, 528)으로 구성되며, 분리된 컴퓨터 시스템 각각은 제1 웹 서비스 인터페이스를 클라이언트 컴퓨터에 제공하고 제2 웹 서비스 인터페이스를 도4를 참고로 하여 언급된 파트너 서비스 제공자에게 제공하는 데이터 볼트 애플리케이션(520)을 동작시킨다. 원격 데이터 저장장치는 중복 파일 저장장치 및 데이터베이스 시스템들(532, 534)을 포함할 수도 있으며, 둘 이상의 지리적으로 분산된 원격 데이터 저장장치 컴퓨터들(526, 528)이 서로 지리적으로 결합되어 있을 수도 있으며, 또는 원격 데이터 저장장치 컴퓨터들(526, 528) 뿐만이 아니라 파트너 서비스 제공자(522) 및 클라이언트(502) 모두에 지리적으로 멀리 떨어져 있을 수도 있다.

<40> 도6a 내지 도6f는 본 발명의 일 실시예를 나타내는 웹 서비스 기반의, 데이터 백업 및 데이터 보존 서비스의 일 양태를 나타낸다. 도6a 내지 도6f는 웹서비스 기반 데이터 백업 및 데이터 보존 서비스의 피처의 도면 부호를 이용한다. 도6a는 클라이언트(602) 및 데이터 볼트(604)를 나타낸다. 전술된 바와 같이, 클라이언트(602)은 클라이언트 컴퓨터에서 동작하는 웹 서비스 기반의, 데이터 백업 및 데이터 보존 서비스 애플리케이션이고, 데이터 볼트(604)는 하나 이상의 원격 데이터 저장장치 컴퓨터에서 동작하는 웹 서비스 기반 데이터 볼트 애플리케이션이다. 웹서비스 애플리케이션이 클라이언트에게 제공한 파일 저장 동작은 저장을 위해 일반적으로 파일 형태로, 클라이언트로부터 데이터 볼트로 데이터 전송을 제공한다. 처음에, 클라이언트는 클라이언트가 데이터 볼트에 백업 또는 보존되기를 원하는 평문 파일(606)을 갖는다. 클라이언트는 또한 오직 자신만이 액세스할 수 있도록 암호 키(608)를 유지한다. 클라이언트는 파트너 서비스 제공자를 통해 데이터 백업 및 데이터 보존 서비스와 계약하고, 데이터 저장동작을 구성한다. 구성의 일부로서, 클라이언트는 데이터 볼트에 의해 장치를 배치한다. 즉, 데이터 볼트의 관점에서, 클라이언트는 장치 식별자를 구비한 원격 장치이다. 데이터 볼트는 원격 장치에 의해 암호화되고 원격 장치(610)에 의해 데이터 볼트로 전송된 파일을 저장한다. 이 파일들은 파일 ID(612)와 같은 파일 ID들과 관련이 있어서 클라이언트에 의한 요청이 있을 경우 데이터 볼트를 나중에 검색하고 저장되고 암호화된 파일을 보내도록 한다.

<41> 따라서 데이터 볼트는 극장, 버스 정류소, 또는 다른 이와 같은 서비스 제공자에 의해 제공되는 의상 체크인 서비스와 유사한 논리적인 서비스를 제공한다. 클라이언트는 하나 이상의 아이템을 검사할 수 있고, 아이템에 대한 식별 태그를 수신한다. 서비스 제공자는 매칭 식별번호를 갖는 태그를 저장된 아이템에 부착한다. 나중에, 클라이언트는 태그를 제출하여 의상의 하나 이상의 아이템을 검색할 수 있고, 서비스 제공자는 이때 저장된 의상과 매치시킨다. 도6a 내지 도6f에서, 저장되고 암호화된 파일은 옷걸이에 매달린 부착된 식별 태그를 갖는 의상의 품목으로 기호로 표시되어 상기에 제시된 아날로그이지만 실제로는 파일서버 또는 어떤 다른 파일 저장장치에 전자적으로 저장되는 파일을 강조한다.

<42> 또한 데이터 볼트는 은행 또는 소매시설의 안전을 목적으로 사용하기 위해 생각할 수 있는 안전한 데이터베이스를 포함한다. 데이터베이스의 한 기능은 클라이언트 암호키(616)의 접근할 수 없는 사본을 안전하게 저장하는 것이다. 만일 여러 이유로, 클라이언트가 암호키(608)를 분실할 경우, 클라이언트는 암호키를 데이터 볼트로부터 얻을 수 있다. 그러나, 데이터 볼트 그 자체는 암호키에 접근할 수 없으므로, 암호화된 파일(610)에 저장된 어떤 정보에도 접근할 수 없다. 클라이언트는 보통 데이터 볼트에 백업되고 보존된 클라이언트가 가진 모든 파일의 로컬 리스트를 보존한다. 그러나, 여러 이유로, 클라이언트가 자신의 파일 리스트를 분실하였을 경우, 클라이언트는 데이터 볼트로부터 파일의 암호화된 리스트를 검색할 수 있으며, 데이터 볼트는 암호화된 파일 속성을 안전한 데이터베이스(614)에 저장한다. 그러나, 데이터 볼트 자신은 데이터 볼트에 저장된 파일 속성 정보에 접근할 수 없다. 따라서, 데이터 볼트에 백업되고 보존된 파일 속성 또는 파일 내용에 관련된 어떤 정보도 클라이언트 컴퓨터를 전혀 떠날 필요가 없다. 데이터 볼트에 백업되고 보존된 모든 데이터는 클라이언트의 데이터를 암호화하기 위해 클라이언트에 의해 사용된 암호화 기술만큼 안전하고, 클라이언트의 컴퓨터에만 접근할 수 있다.

<43> 클라이언트는 데이터 볼트내에 평문 파일(606)을 저장하기 위해 두 개의 다른 동작을 실행한다. 먼저, 도6b에 나타난 바와 같이, 클라이언트는 결합한 파일을 이어서 저장하려고 하는 데이터 볼트에 암호화된 파일 속성과 함께 클라이언트 장치 번호를 발송한다. 그 다음, 도6c에 나타난 바와 같이, 데이터 볼트는 파일 ID를 클라이언

트에게 보내는데, 데이터 볼트는 클라이언트가 저장하고자 하는 파일과 결합한다. 다시, 상기에 나타난 바와 같이, 속성은 데이터 볼트에 전송되기 전에 클라이언트에 의해 암호화된다. 따라서, 데이터 볼트는 암호화된 속성(616)을 안전한 데이터베이스 내에 저장하며, 데이터 볼트 자신은 암호화된 속성에 접근하거나 암호화된 속성을 판독할 수 없다. 도6d에 나타난 바와 같이, 저장될 파일에 대해 수신된 파일 ID를 갖고, 클라이언트는 평문 파일(618)의 암호화된 버전을 생성하기 위해 평문 파일을 암호화한다. 그 다음에, 도6e에 나타난 바와 같이, 클라이언트는 데이터 볼트에 의해 클라이언트에게 이미 보낸 클라이언트의 장치 번호 및 파일 ID와 함께 암호화된 파일을 데이터 볼트에 저장하기 위해 발송한다. 마지막으로, 도6f에 나타난 바와 같이, 암호화된 파일(618)을 파일 ID(620)와 결합하여 클라이언트 컴퓨터에 결합한 장치에 배치된 파일 저장장치에 저장하여, 데이터 볼트에 백업되고 보존된 파일을 식별하는 국부적으로 저장된 정보를 분실할 경우, 데이터 볼트로부터 연속적으로 파일을 검색하여 사용하려는 데이터 볼트로부터 클라이언트의 장치에 결합된 암호화된 파일 속성/파일 ID 쌍을 요청할 수 있다.

<44> 도7a 내지 도7b는 클라이언트를 초기에 구성하는 것뿐만 아니라 웹 서비스 기반의 데이터 백업 및 데이터 보존 서비스에 대해 계약하려는 클라이언트와 파트너 서비스 제공자 사이의 상호작용을 예시한다. 도7a 내지 도7b는 클라이언트, 파트너 서비스 제공자, 및 웹 서비스 프로토콜을 따르는 데이터 볼트 사이의 데이터 전송을 예시한다. 도7a 내지 도7b에서, 클라이언트, 파트너 서비스 제공자 및 데이터 볼트를 왼쪽에서 오른쪽으로 순서대로 나타내는 세 개의 열이 나타나 있다. 도7a에 나타난 바와 같이, 클라이언트는 서비스 요청(702)을 파트너 서비스 제공자에게 전송하고, 파트너 서비스 제공자는 클라이언트에게 클라이언트측의 웹서비스 기반 데이터 백업 및 데이터 보존 서비스 애플리케이션 소프트웨어(704)를 보내준다. 이 거래가 도7a의 두 개의 메시지를 통해 발생하는 것처럼 보이지만, 이 거래는 비교적 긴 프로토콜을 포함할 수도 있으며, 이 프로토콜에서 클라이언트는 파트너 서비스 제공자에 의해 제공된 웹 페이지에 처음에 응답하고, 다양한 형식 및 지불 정보을 수신하고 기입하여 돌려보내며, 웹 서비스 기반 데이터 백업 및 데이터 보존 서비스에 대해 성공적으로 계약하고 클라이언트측 소프트웨어를 수신하도록 임의의 추가된 거래 관련 동작을 실행할 수도 있다.

<45> 일단 클라이언트측 소프트웨어가 설치되면, 공용/개인 암호키 쌍이 클라이언트 측 컴퓨터에 생성되고, 공용/개인 암호키 쌍의 공용 암호키(706)는 클라이언트에 의해 파트너 서비스 제공자에게 새로운 구성에 대한 요청의 일부로써 전송된다. 그 다음에, 파트너 서비스 제공자는 클라이언트의 공용암호키를 새로운 장치 요청(708)과 함께 데이터 볼트로 전송한다. 데이터 볼트는 파트너 서비스 제공자와 클라이언트 모두를 대신하는 새로운 장치를 생성하고, 응답 메시지(710) 내에서 클라이언트의 공용키를 사용하여 장치 구성 정보를 암호화하고, 응답 메시지를 파트너 서비스 제공자에게 보낸다. 파트너 서비스 제공자는 응답 메시지 내에서 암호화된 장치 구성 정보를 포함하고, 또한, 클라이언트(712)를 대신하여 파트너 서비스 제공자에 의해 생성된 패스프레이즈(passphrase)를 응답 메시지에 포함하고, 응답 메시지를 클라이언트에게 보낸다. 응답 메시지(712)를 수신하면, 클라이언트는 파트너 서비스 제공자에 의해 공급된 패스프레이즈를 암호화된 장치 구성 정보와 함께 얻어내고, 고객의 개인 암호키를 사용하여 장치 구성 정보를 해독할 수 있다. 본 발명의 일부 실시예에서, 클라이언트는 파트너 서비스 제공자에 의한 패스프레이즈 생성에 의존하기보다는 하나 이상의 패스프레이즈를 선택 또는 제시할 수 있다. 그 다음에 장치 구성 정보는 클라이언트 측 소프트웨어에 의해 연속적인 데이터 백업 및 데이터 보존 동작에 대한 클라이언트측 애플리케이션을 충분하게 구성하기 위해 사용된다. 파트너 서비스 제공자는 데이터 볼트에 의해 클라이언트로 보내진 장치 구성 정보를 가로채거나 접근하거나 사용할 수 없는데, 파트너 서비스 제공자가 클라이언트의 개인 암호키를 갖지 않기 때문이다. 파트너 서비스 제공자에 의해 클라이언트로 보내진 패스프레이즈도 데이터 볼트에 사용할 수 없다는 것을 유념하라. 그러나, 본 발명의 많은 실시예에서, 파트너 서비스 제공자는 클라이언트를 대신하여 생성된 패스프레이즈와 함께 에스크로우(escrow) 서비스를 사용하여 파트너 서비스 제공자의 개인 암호키를 저장하는 것에 동의하여, 파트너 서비스 제공자가 동작을 지속하지 않거나 파트너 서비스 제공자를 통해 계약된 데이터 볼트로부터 데이터 백업 및 데이터 보존 서비스 갖는 클라이언트 컴퓨터에 사용할 수 없는 경우에, 패스프레이즈 또는 클라이언트에게 제공된 패스프레이즈가 클라이언트에 의해 회복되고, 파트너 서비스 제공자의 개인 암호키는 데이터 볼트에 의해 회복될 수 있다.

<46> 다음으로, 도7b에 나타난 바와 같이, 클라이언트는 클라이언트의 컴퓨터에만 알려진 새로운 암호키를 생성하고 파트너 서비스 제공자에 의해 제공된 패스프레이즈를 사용하여 새로운 암호키를 암호화하여 새로운 패스프레이즈 암호화 암호키(714)를 만들어낸 다음, 파트너 서비스 제공자의 공용 암호키를 사용하여 새로운 패스프레이즈 암호화된 새로운 암호키를 암호화하여 이중으로 암호화된 새로운 암호키(716)를 만들어내며, 이중으로 암호화된 새로운 암호키(716)는 클라이언트에 의해 데이터 볼트로 발송된다. 데이터 볼트는 이중으로 암호화된 새로운 클라이언트 암호키를 클라이언트를 위해 할당된 장치에 대한 장치 식별자와 함께 안전한 데이터베이스에 저장한다. 저장된, 이중으로 암호화된 클라이언트 암호키는 도7a에 안전한 암호키(716)로 표시된다. 데이터 볼

트는 승인 메시지(718)를 클라이언트 컴퓨터로 역 발송한다. 본 발명의 일부 실시예에서, 승인은 파트너 서비스 제공자를 거쳐서 발송될 수도 있다.

<47> 클라이언트 컴퓨터는 이어지는 데이터 백업 및 데이터 보존 동작을 위해 완전하게 설정된다. 클라이언트 컴퓨터는 국부적으로 저장된 클라이언트 컴퓨터의 암호키의 사본을 갖고, 그 다음에 클라이언트 컴퓨터는 백업 또는 보존을 위해 데이터 볼트에 전송된 모든 데이터를 암호화하는데 사용된다. 오직 클라이언트만 클라이언트의 암호키를 알고, 클라이언트의 암호키가 데이터 볼트 내에서 이중으로 암호화되었기 때문에, 데이터 볼트 또는 파트너 서비스 제공자는 모두 데이터 볼트 내에 클라이언트에 의해 저장된 정보를 해독하기 위해 클라이언트의 암호키에 접근할 수 없다. 이의 중요한 결과는 클라이언트 데이터가 데이터 볼트에서 안정할 뿐만 아니라 저장된 파일과 관련된 파일 속성이 안전해서, 파트너 서비스 제공자 뿐만 아니라 데이터 볼트도 판독할 수 없고, 또는 저장된 데이터 속성에 접근할 수 없다는 것이다. 예를 들어, 법률사무소는 법률 사무소의 클라이언트를 연성시키거나 다양한 법률상의 문제 또는 법률 사무소에 의해 법률사무소를 대신하여 처리된 거래들을 연상시키는 파일이름으로 많은 파일을 저장할 수도 있다. 이러한 파일의 내용이 데이터 볼트 또는 파트너 서비스 제공자에 접근할 수 없고, 접근할 수 있는 파일 이름일지라도, 많은 비밀 정보가 데이터 볼트 또는 파트너 서비스 제공자에 대한 접근권을 얻는 데이터 볼트, 파트너 서비스 제공자 또는 악성의 써드파티(third party)에 의해 수집될 수도 있다. 그러나, 본 발명의 실시예를 나타내는 웹 서비스 프로토콜 하에서, 파일 이름, 파일 소유자 및 다른 파일 속성이 클라이언트 컴퓨터를 떠나기 전의 암호화에 의해 완전하게 보호된다.

<48> 도8은 도6a 내지 도6f에 대해 전술된, 데이터 볼트 내에 데이터를 안전하게 저장하기 위해 실시된 클라이언트 측 동작을 예시하는 간단한 흐름 제어 프로그램이다. 단계 802에서, 클라이언트는 파일에 관한 파일 속성을 암호화하고 저장 요청을 데이터 볼트에 직접 발송한다. 저장 요청은 클라이언트 측 소프트웨어의 초기 구성의 일부로써 수신된 클라이언트 컴퓨터에 관한 장치에 대한 장치 ID를 포함한다. 단계 804에서, 그 대신에 클라이언트는 데이터 볼트로부터 파일 ID를 수신한다. 단계 806에서, 클라이언트는 파일을 저장할 수 있도록 암호화하고 암호화된 파일을 파일 ID와 함께 데이터 볼트로 발송한다. 단계 808에서, 클라이언트는 데이터 볼트로부터 파일이 하나 이상의 원격 데이터 저장장치에 성공적으로 저장되었다는 승인을 수신한다.

<49> 도9는 클라이언트를 대신하는 파일 저장에 관하여 데이터볼트에 의해 수행되는 동작을 예시한다. 인증 및 검증(validation) 단계 수행 후, 데이터 볼트는 클라이언트를 대신하여 새로운 파일 ID를 생성한다. 단계 906에서, 데이터 볼트는 단계 902에서 수신된 저장 요청으로부터 파일 속성을 추출하고 새롭게 생성된 파일 ID와 함께 암호화된 파일 속성을 안전한 데이터베이스에 저장한다. 단계 908에서, 데이터 볼트는 파일 ID를 클라이언트에게 보낸다. 단계 910에서, 데이터 볼트는 파일 ID와 함께, 데이터 볼트에 저장하기 위해 암호화된 파일을 수신하고, 단계 912에서, 데이터 볼트는 파일 ID에 관련된 데이터베이스 엔트리의 암호화된 파일의 수신에 주목하고, 암호화된 파일을 파일 서버 또는 다른 데이터 저장 장치에 저장하고, 클라이언트에게 승인을 발송한다.

<50> 클라이언트에 의한 파일 검색은 상대적으로 단순하고, 데이터 볼트에 파일 을 요청하는 것을 포함하고, 파일 ID 및 장치 식별자를 수반하며, 일단 요청이 데이터 볼트에 의해 적절하게 검증 및 인증되면, 데이터 볼트는 지정된 파일을 파일 서버에 정하고, 지정된 파일을 클라이언트에게 보낸다. 전술된 바와 같이, 클라이언트는 암호화된 파일 속성/파일 ID 쌍을 데이터 볼트에 요청하고 데이터 볼트로부터 수신하며, 클라이언트는 이 정보의 국부적인 사본을 언젠가는 잃게 될 것이다. 또한, 클라이언트는 이중으로 암호화된 클라이언트 암호키를 데이터 볼트에 요청 및 데이터 볼트로부터 수신하고, 클라이언트 암호키를 언젠가는 잃게 될 것이다. 이 요청은 파트너 서비스 제공자를 통해 클라이언트에 의해 만들어지며, 이 경우 파트너 서비스 제공자는 패스프레이즈 암호화된 암호키를 클라이언트에게 전송하기 전에 클라이언트를 대신하여 이중 암호의 제1 레벨을 해독할 수 있다.

<51> 본 발명의 다른 실시예에서, 웹 서비스 기반 데이터 백업 및 데이터 보존 서비스는 추가 서비스를 제공할 수도 있다. 예를 들어, 본 발명의 일 실시예에서, 파일 암호화 이전에, 클라이언트 측 애플리케이션은 로컬 인덱스에 저장된 파일에 대한 인덱스 정보를 생성하여, 클라이언트 측 애플리케이션이 텍스트 문자열 또는 다른 검색 정보에 대해 원격으로 저장된 파일을 검색하도록 한다. 바꿔 말하면, 국부적으로 저장된 인덱스는 단어 인덱스, 또는 다른 데이터 객체 인덱스를 포함하고, 여기에서 단어는 파일 ID에 관한 것이다. 그 다음, 파일이 암호화될 경우, 파일에 대해 생성된 인덱스 정보도 암호화되고 저장을 위해 개별적으로 데이터 볼트로 발송될 수도 있다. 국부적인 인덱스 정보가 클라이언트에 의해 언젠가 분실될 경우, 인덱스 정보는 데이터 볼트로부터 암호화된 형태로 검색될 수 있다.

<52> 본 발명의 다양한 실시예에 의해 제공된 추가 서비스는 효과적인 파일 업데이트이다. 다양한 실시예에서, 클라이언트는 데이터 볼트의 저장장치에 대한 업데이트 파일을 클라이언트에 국부적으로 저장된 업데이트된 파일의

이전 버전과 비교하고, 두 파일 사이의 차이를 계산한다. 그 다음에, 차이를 설명하는 메타데이터와 함께 클라이언트는 오직 차이만을 암호화하고, 저장을 위해 업데이트된 전체 파일을 전송하기보다는 이러한 암호화된 차이와 메타데이터를 데이터 볼트로 전송한다. 데이터 볼트는 전체 업데이트된 것과 함께 파일의 제1 버전을 저장할 수 있고, 저장되고 업데이트된 파일이 클라이언트에 의해 연속적으로 요청될 경우, 첫 번째의 완벽한 버전과 그 다음의 업데이트 모두를 클라이언트에게 보낼 수 있거나, 가장 최근의 업데이트 파일과 이전에 요청된 업데이트 파일을 클라이언트에게 보낼 수 있다. 전체 업데이트된 파일 대신 업데이트된 차이의 암호화 및 전송이 계산상으로 효과적일 뿐만 아니라 전송 및 저장에도 더욱 효과적이다.

<53> 본 발명의 다양한 실시예의 더욱 상세한 설명

<54> 아래에는, 본 발명의 구현된 실시예의 더욱 상세한 설명이 제공된다. 도10은 전체 레벨에서 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템의 클라이언트 측 및 서버 측 부분을 예시한다. 백업-복원 및 보존 시스템(1002)의 클라이언트 측 부분은 다수의 사용자 장치, 일반적으로는 개인용 컴퓨터(PC)를 포함한다. 서술된 실시예에서, 백업, 복원 및 보존 서비스가 클라이언트 장치의 입도(granularity)에서 백업-복원 및 보존 시스템의 서버측 부분에 의해 제공된다. 바꿔 말하면, 백업, 복원 및 보존 서비스는 개인용 컴퓨터와 같은 물리적인 하드웨어 장치에 제공된다. 다른 실시예에서, 백업, 복원 및 보존 서비스는 하드웨어 장치의 개별 사용자 부분과 같은 입도의 아주 미세한 레벨에서 제공될 수도 있다.

<55> 인터넷 프로토콜(1006) 상에 구현된 안전한 소켓층(secure socket layer, SSL)을 사용하는 일부 실시예에서, 클라이언트 장치는 안전한 연결부를 거쳐 백업-복원 및 보존 시스템(1004)의 서버측 부분과 통신한다. 백업-복원 및 보존 시스템의 서버측 부분(1004)은 하나 이상의 웹 서버(1008), 하나 이상의 공유 디스크 서버(shared-disk server, 1010), 하나 이상의 작업 서버(job server, 1012), 하나 이상의 데이터베이스 서버(1014), 하나 이상의 액티브 디렉토리 서버(active directory server, 1016), 하나 이상의 영구 데이터 저장장치(permanent data storage device, 1018) 및 작동 모니터(operations monitor, 1020)를 포함한다. 하나 이상의 웹 서버(1008)는 클라이언트 장치(1002)와 직접 상호작용하기 때문에, 웹 서버는 클라이언트 장치 및 다른 장치로부터 백업-복원 및 보존 시스템(1004)의 서버측 부분 내에서 방화벽(1022~1024)에 의해 분리되어 있다.

<56> 도10에 나타난 백업-복원 및 보존 시스템(1004) 전체는 본 발명에 따르는 백업-복원 및 보존 시스템의 무수하게 가능한 구성 중 하나만을 예시한다. 백업-복원 및 보존 시스템의 극단적인 전체 서버 측 부분은 단일 서버 컴퓨터에 구현될 수도 있고, 다른 극단적이고 복잡한 다중 구성요소 서버측 부분은 국부적이고 지리적으로 복제되어 단점 및 피해에 대한 극도로 높은 수준의 내성 및 높은 유용성을 제공할 수도 있다. 도11은 전체 레벨에서, 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템의 서버측 부분의 단일 서버 구현을 예시한다. 단일 서버 구현은 단일 서버 컴퓨터(1108) 내의 백업-복원 및 보존 시스템의 서버측 부분의 기능성을 함께 제공하는 인터넷 정보 서버 애플리케이션(1102), 액티브 디렉토리(active directory, 1104) 및 SQL 서버를 포함한다. 도12는 본 발명의 다른 실시예를 나타내는 복잡하고 복제된 백업-복원 및 보존 시스템을 예시한다. 도12에 나타난 다중 구성요소의 복제된 백업-복원 및 보존 시스템에서, 분배 파일 시스템 분류법(1210)에 의해 구현된 웹 서버(1204)의 뱅크를 포함하는 제1 데이터 센터(1202), 데이터베이스 서버(1206), 액티브 디렉토리 서버(1208) 및 영구 데이터 저장장치는 글로벌 로드 발란서(1212) 및 로컬 로드 발란서(1214)를 거쳐 클라이언트 컴퓨터로부터 인터넷을 거쳐 접근될 수도 있다. 따라서, 도12에 나타난 실시예에서, 백업-복원 및 보존 시스템의 두 개의 분리된 다중 구성요소 서버 측 부분이 높은 유용성뿐만 아니라 단점 및 피해에 대한 내성을 제공하기 위해 공존한다. 또 다른 실시예에서, 다중 구성요소 서버측 부분은 3중, 4중 또는 더 여러 번 복제될 수도 있다. 또한, 높은 단의 구현에서, 각 서버측 부분의 영구 데이터 저장장치는 RAID-5 및 RAID-6 저장 시스템에서 발견되는 오류 코드 부호화 기반 중복을 포함하는 중복성 도입 기술의 다른 유형에 의해 반영되거나 여유 있게 저장될 수도 있다.

<57> 도13a 내지 도13c는 도10의 전체 레벨에서 예시된 백업-복원 및 보존 시스템 내의 기본적인 기능성을 예시한다. 도13a는 클라이언트 장치(도10의 1002) 내의 기본적인 기능성을 예시한다. 백업-복원 및 보존 시스템의 클라이언트 장치 부분은 세 개의 상이한 프로세스를 포함한다. 제1 프로세스는 클라이언트 장치의 터미널(1306)에 표시된 아이콘(1304)을 통한 쌍방향 호출을 포함하는 임의의 다양한 루틴 호출 방법을 거쳐 사용자에 의해 요구된 사용자 인터페이스 루틴(1302)으로서 구현된다. 사용자 인터페이스 루틴은 백업-복원 및 보존 시스템에 의해 연속적이고 자동으로 백업될 클라이언트 장치 내의 파일 리스트(1312) 및 다른 파일 같은 객체를 포함하는, 다른 것들 중에서 국부적으로 저장된 카탈로그(1310)를 사용자가 수정하는 것을 허용하는 기본적인 사용자 관리 및 사용자 구성 서비스를 제공한다. 카탈로그(1310)는 각 파일에 대한 파일 변조 검출 기간, 정정횟수의 표시, 또는 주어진 파일의 예, 또는 유지하려는 파일과 같은 개체, 파일 또는 파일과 같은 개체에 대해 사용자가 요구하

는 보호 수준의 표시 및 다른 이러한 파라미터와 같은 구성 정보를 추가로 포함할 수도 있다.

<58> 사용자는 사용자 인터페이스 루틴(1302)에 클라이언트 장치 디스플레이 모니터(1306)에 디스플레이된 그래픽 사용자 인터페이스를 통해 다양한 형태의 명령을 발행한다. 명령은 파일 및 파일과 같은 개체를 백업 리스트로부터 삭제 또는 추가하고, 하나 이상의 개별 파일을 특정의 이전에 백업된 예를 복원하기 위한 명령, 백업-복원 및 보존 시스템 내에 저장된 정정 이력을 삭제하기 위한 명령 및 다양한 다른 실시예에서 다양한 추가 명령을 포함한다.

<59> 두 개의 추가 프로세스(1314, 1316)는 클라이언트 장치 내에서 윈도우 서비스로써 계속 작용한다. 제1 윈도우 서비스 프로세스(1314)는 백업 및 복원 동작을 실행에 대한 책임이 있는 주고객 측 서비스 프로세스이다. 제2의 연속적으로 실행되는 윈도우 서비스 프로세스(1316)는 백업-복원 및 보존 시스템 파트너와 함께 백업-복원 및 보존 시스템의 서버 측 부분과 데이터를 교환하기 위해 안전한 소켓 층(SSL) 및 배경 지능 전송 서비스(BITS)를 이용하는 운송 서비스이다. BITS는 원격의 엔티티와 교환하기 위해 배경 프로세스로써 예비 네트워크 대역폭 및 프로세싱 주기를 사용한다.

<60> 주고객 측 프로세스(1314)는 카탈로그(1310)에 저장된 데이터에 의해 한정된 것처럼 백업-복원 및 보존 시스템에 의해 연속적으로 백업된, 파일(1320)과 같은, 각 파일 또는 파일 유사 개체를 주기적으로 점검하는 감시 기능(1318)을 포함한다. 감시 프로세스(1318)는 최근의 주기적인 감시 주기 이후에 파일이 변경될지 여부를 이전에 기록된 타임 스탬프와 현재 파일 타임스탬프의 비교 또는 다른 이와 같은 정보에 기반하여 판단한다. 파일이 변경되었을 경우, 백업 루틴(1322)은 변경된 파일 및 파일(1324) 자신을 사용하고 백업-복원 및 보존 시스템의 국부적으로 저장된 정보 또는 서버 측 부분으로부터 획득한 정보 중 하나인 이전의 파일의 인스턴스 사이의 블록 차이를 계산한다. 블록 차이는 영구적인 저장을 위해 변경되고 백업-복원 및 보존 시스템의 서버 측 부분에 전송될 필요가 있는 파일의 이러한 부분을 포함하기 위해 결정된 Δ블록 셋이다. 백업-복원 및 보존 시스템의 서버측 부분에 의해 일반적으로 저장되지 않는 것으로 알려진 Δ블록 또는 Δ블록 서브 셋이 로컬 캐시(1328)와 업로드 파일(1326)에 추가된다. 업로드 파일(1326)은 백업-복원 및 보존 시스템의 서버측 부분에 전송 서비스 프로세스(1316)에 의해 하나씩 차례로 전송되는 업로드 파일(1328)의 큐로 배열된다. 사용자가 사용자 인터페이스 루틴(1302)을 통해 복원을 요청할 경우, 주 고객측 프로세스(1314) 내의 복원 프로세스는 로컬 캐시(1328) 및 파일(1322)의 임의의 기존 부분으로부터 국부적으로 획득될 수 있는 파일의 블록을 결정하게 하고, 전송서비스 프로세스(1316)를 거쳐 백업-복원 및 보존 시스템의 서버측 부분으로부터 모든 다른 필요한 블록을 검색하고, 검색된 블록 및 국부적으로 사용가능한 블록을 사용하여 파일의 복원된 버전(1332)을 모은다.

<61> 도13a에 예시된 백업-복원 및 보존 시스템의 고객측 부분은 많은 다양한 가능한 구현 중 하나 일 뿐이다. 백업, 보존, 카탈로그 캐시, 전송 및 사용자 인터페이스 상관관계는 몇 개의 모듈 및 프로세스로 함께 결합될 수도 있거나, 수많은 상이한 기능의 모듈, 프로세스 및 서비스로 선택적으로 분리될 수도 있다.

<62> 도13b 내지 도13c는 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템(도10의 1004)의 서버측 부분의 기능 동작을 예시한다. 도13b에 나타난 바와 같이, 하나 이상의 웹 서버(1008) 각각은 백업-복원 및 보존 시스템이 백업, 복원 및 보존 서비스를 제공하는 크라이언트 장치 각각의 전송 서비스 프로세스(도13a의 1316)와 인터넷 같은 SLL 층(1342) 및 통신 매체를 거쳐 통신하는 라우팅 관리자 기능성(1340)을 포함한다. 라우팅 관리자(1340)는 하나 이상의 데이터베이스 서버(1014) 중 하나에 액세스하여 클라이언트 장치가 웹 서버(1008)로의 SSL 접속을 연 후에 클라이언트 장치에 의해 공급된 신임장을 서버측 부분 액티브 디렉토리 서버(104)에 의해 저장된 신임장에 부합시킨다. 하나 이상의 데이터베이스 서버(1014)는 광범위한 카탈로그(1346), 파트너 및 파트너 암호키(1348)에 관해 저장된 정보, 클라이언트 장치에 관해 저장된 정보, 클라이언트 장치에 관련된 신임장, 클라이언트 장치(1350)에 대해 조건부로 날인된(escrowed) 파일 암호키 및 다른 정보를 나타내는 다양한 데이터베이스 테이블 또는 파일에 정보를 저장하고, 여기에서 정보를 검색하고, 관리하는 메타 데이터 관리자(1344)를 실행시킨다. 라우팅 관리자가 연결되는 클라이언트 장치를 확인할 경우, 라우팅 관리자는 클라이언트 장치로부터 업로드 파일을 수신할 수 있으며, 업로드 파일을 하나 이상의 공유된 비휘발성 저장 서버(1010)에 의해 공급된 공유된 비휘발성 저장장치(1352)의 업로드 파일을 기다린다. 또한, 라우팅 관리자는 수신된 업로드 파일에 대한 정보를 하나 이상의 데이터베이스 서버(1014) 중 하나에서 실행하는 메타-데이터 관리자(1344)를 전송하여, 메타-데이터 관리자가 작업 요청을 작업 큐에 입력할 수 있게 한다. 유사하게, 다운로드 파일(1356)과 같은 다운로드 파일은 클라이언트 장치에 라우팅 관리자(1340)에 의한 전송을 위해 공유된 비휘발성 저장장치(1352) 내의 다운로드 파일 큐에 저장될 수도 있다.

<63> 클라이언트 장치 및 라우팅 관리자 사이에 전달된 데이터 벌크가 개별 파일 또는 파일 유사 개체를 저장하기 위

해 클라이언트 장치에 필요한 백업 데이터를 나타내는 업로드 파일과 암호화된 데이터 블록을 나타내는 다운로드 파일을 구성하면, 라우팅 관리자(1340)는 클라이언트 장치 및 다른 구성 및 관리 명령을 위해 카탈로그 데이터에 업데이트를 포함하는 명령과 같은 추가 명령을 수신할 수 있다. 많은 경우에, 이 명령들은 라우팅 관리자에 의해 데이터베이스 서버에서 실행되는 메타-데이터 관리자에게 직접 전달되고, 메타-데이터 관리자를 사용하여 즉시 명령을 실행하고 라우팅 관리자에게 응답을 보내고 후속 프로세싱을 위한 작업 대기열내의 명령을 기다린다. 하나 이상의 액티브 디렉토리 서버(1016)는 분배 네트워트 영역 내의 네트워크 개체를 관리하는 책임이 있다. 구성요소 시스템, 데이터 리소스 및 다른 개체에 의해 실행되는 서비스 및 애플리케이션을 포함하는 백업-복원 및 보존 시스템의 서버측 부분의 다양한 구성요소 시스템은 하나 이상의 액티브 디렉토리 서버에 의해 생성되고 관리된 글로벌 이름 공간(global name space)을 통해 접근가능하다.

<64> 도13c에 나타난 바와 같이, 하나 이상의 작업 서버(1012) 각각에서 동작하는 워크호스 루틴(1360)은 백업 및 복원 동작을 수행할 책임이 있다. 워크호스 루틴(1360)은 데이터베이스 서버(1014)의 컬렉션에 의해 관리되고, 데이터베이스 서버 컬렉션에 저장되는 작업 큐(1354)로부터 연속적인 작업을 디큐하고(de-queue), 디큐된 작업 큐 엔트리에 의해 나타나는 임무를 수행한다. 백업 임무를 위해, 워크호스 루틴(1360)은 개별 작업 큐 엔트리(1364)에 대응하고, 메타-데이터 관리자(1344)에 의해 공급된 메타 데이터를 사용하는 업로드 파일(1362)을 검색하고, 업로드 파일을 파일 서명 시리즈 및 암호화된 데이터 블록으로 분리하고, 파일 서명을 메타-데이터 관리자(1344)를 거쳐 데이터베이스(1014)에 저장하고, 파일 서명과 관련된 암호화된 데이터 블록을 하나 이상의 영구 데이터 서버(1018)에 의해 제공된 영구 데이터베이스에 저장한다. 유사하게, 워크호스 루틴(1360)은 필요한 암호화된 데이터 블록, 저장된 파일 서명 및 작업 큐(1354)로부터의 복원 작업 디큐잉시 영구 데이터 스토어(1018) 및 데이터베이스(1014)를 검색하고, 정보를 다운로드 파일(1366)에 결합하고, 결과적으로 라우팅 관리자(1340)에 의해 클라이언트 장치로 전송하기 위해 다운로드 파일을 공유된 비휘발성 저장장치(1352) 내의 다운로드 파일의 대기열로 큐한다.

<65> 따라서, 도10을 다시 보면, 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템은 백업-복원 및 보존 시스템의 서버측 부분(1004)에 의한 백업, 복원 및 보존 서비스를 제공한 매우 많은 수의 클라이언트 장치를 잠재적으로 포함한다. 백업-복원 및 보존 시스템의 서버측 부분(1004)은 다수의 특수화된 서버 및 컴퓨터 시스템을 포함하는 단일 데이터 센터의 단일 서버 컴퓨터 내에 구현될 수도 있으며, 또는 다수의 복사된 다중 구성요소 데이터 센터로써 구현될 수도 있다. 백업-복원 및 보존 시스템의 서버측 부분은 클라이언트 장치로부터의 요청 수신 및 라우팅할 책임이 있는 클라이언트 장치로의 웹 인터페이스를 포함하며, 웹 인터페이스는 백업-복원 및 보존 시스템의 서버측의 적절한 구성요소를 실행을 위해 요청하고, 백업-복원 및 보존 시스템의 서버측 부분으로부터 클라이언트 장치로 거꾸로 응답 및 데이터를 발송한다. 백업-복원 및 보존 시스템의 서버측 부분은 작업 서버(1012)로의 업로드 파일 및 작업 서버(1012)로부터 웹 인터페이스(1008)로의 다운로드 파일과 통신하고 일시적으로 저장하기 위해 사용되는 공유된 비휘발성 저장공간(1010)을 포함한다. 백업-복원 및 보존 시스템의 서버측의 영구 데이터 저장부분(1018)이 파일 및 파일 유사 개체를 복원하기 위해 클라이언트 장치에 의해 필요할 수도 있는 암호화된 데이터 블록을 저장하는 것과 동시에, 백업-복원 및 보존 시스템(1014)의 서버측의 데이터베이스 부분은 클라이언트 장치의 상태를 따라가기 위해 필요한 메타데이터 및 백업-복원 및 보존 시스템에 의해 실행될 필요가 있는 현재 임무를 저장한다.

<66> 도14a 내지 도14d는 대응 파일 서명 및 암호화된 데이터 블록을 생성하기 위해 파일 및 파일 유사 개체의 주 서비스 프로세스(1314)에 의한 프로세싱을 예시한다. 도14a에 나타난 바와 같이, 파일 또는 파일 유사 개체(1420)는 바이트, 위드, 롱 워드 또는 다른 기본적인 데이터 단위의 순열처럼 보인다. 클라이언트 장치의 주 서비스 프로세스에 의해 실행된 제1 단계에서, 클라이언트 장치(1420) 내에 존재하는 파일 또는 파일 유사 개체는 논리적으로 자연 블록으로 분할된다. 자연 블록 경계는 도14a에 점수직선(1404) 같은 점선의, 수직선으로 표시된다. 자연 블록은 가변 길이를 갖고, 자연 블록 경계는 시간에 따라 증가하는 파일 변조에 대해 서로 상대적으로 독립적일 수도 있는 파일의 부분을 분리하는 파일 내의 경계에 대응한다. 바꿔 말하면, 동작 및 다른 파일 동작을 편집하여 시간에 따라 파일이 변조되면, 다수의 인트라 블록 변화가 다수의 인트라 블록 변화보다 더 자주 발생해야하기 때문에, 상대적으로 동시의 변화 셋이 자연블록 내에 배치된다. 그러나, 가변 길이 블록 방법은 파일 또는 파일 유사 개체 내의 자연블록의 최적 측정을 나타내지만, 파일 변조에 관해서는 파일을 독립 블록으로 정확하게 분할하는 것을 보장하지 못한다.

<67> 다음 단계에서, 상대적으로 작은 자연 블록은 일반적으로 순차적으로 함께 모아지며 고정된 길이 블록(1408)과 같은 거의 고정된 길이의 블록으로 연속적으로 구성된다. 예를 들어, 거의 고정된 길이의 블록(1408)은 제1 단계에서 식별된 제1의 4개의 자연블록(1406, 1410-1412)을 포함한다. 자연 블록의 다음 셋(1414-1417)은 다음의

고정된 길이 블록(1410) 안으로 결합된다. 따라서, 파일 처리 방법의 제1의 두 단계의 결과로서, 파일 또는 파일 유사 개체는 연속적으로 배열되고, 증가의 변화에 관해 서로 합리적으로 독립되도록 평가되는 거의 고정된 길이 블록의 셋으로 분할된다. 거의 고정된 길이의 블록은 각각의 거의 고정된 길이 블록에 결합된 자연 블록 길이의 합의 불일치로 인해 길이가 약간 변할 수도 있다. 본 발명의 일 실시예에서, 거의 고정된 길이 블록은 64K 바이트에 가까운 길이를 갖는다.

<68> 도14b에 나타난 바와 같이, 블록 해시는 각각의 거의 고정된 길이 블록에 대해 계산된다. 본 발명의 일 실시예에서, 클라이언트 장치 파일 암호키(1420), 압축 알고리즘 식별자(1422) 및 암호 알고리즘 식별자(1424)는 거의 고정된 길이 블록(1426) 내의 데이터와 결합하고 MD5 해시 기능(1428)과 같은 암호 해시 기능에 의해 처리되어, 블록 해시(1430)를 생성한다. 파일 암호키 포함, 압축 알고리즘 ID 및 암호 알고리즘 ID는 파일 암호키, 압축 알고리즘 또는 암호 알고리즘이 이 클라이언트에 의해 변경되어야 하고, 새로운 암호키 및/또는 압축 알고리즘에 의해 암호화되고 압축된 블록은 이전에 사용된 암호키, 암호 알고리즘 및/또는 압축 알고리즘에 의해 암호화 및/또는 압축된 블록으로부터 쉽게 구별될 수 있다는 것을 보증한다. 또한, 파일 암호키의 포함은 백업-복원 및 보존 시스템의 서버측 부분에 관한 특정 형태의 보안 공격을 무디게 할 수 있다. 블록 해시(1430)는 수치적인 합계 또는 원래의 거의 고정된 길이 블록(1426)의 개요로 생각될 수 있다. 일반적으로, 블록 해시는 예를 들어, 256 바이트, 512 바이트, 1024 바이트 또는 다른 2 바이트의 배수의 고정된 길이를 갖는다. 암호화 해시 기능의 사용은 클라이언트 장치에 존재하는 임의의 파일 또는 파일 유사 개체로부터 임의의 클라이언트 장치에 의해 생성된 두 개의 상이한 거의 고정된 길이 블록이 동일한 블록 해시를 갖는 기회가 극도로 작은 것을 보장한다. 바꿔 말하면, 블록 해시는, 극도로 높은 가능성으로, 백업-복원 및 보존 시스템을 전체에 거의 고정된 길이 블록의 고유 식별자가 되도록 보장된다.

<69> 도14c에 나타난 바와 같이, 블록에 대응하는 블록 해시의 계산에 이어, 각각의 거의 고정된 길이 블록은 압축 알고리즘(1432)에 의해 압축된 다음 클라이언트의 파일 암호키를 사용하여 암호 알고리즘(1434)에 의해 암호화된다. 이러한 단계는 일반적으로 이전에 계산된 블록 해시(1430)에 의해 식별된 원래의 거의 고정된 길이 데이터 블록(1426)에 대응하는 작고 암호화된 데이터 블록(1436)을 생성한다.

<70> 도14d는 클라이언트 장치의 주 서비스 프로세스에 의한 파일 서명의 계산을 예시한다. 도14a 내지 도14c를 참고로 하여 전술된 바와 같이, 파일 또는 파일 유사 개체는 거의 고정된 길이 블록(1440-1446)의 순서로 먼저 분할된다. 도14b에 나타나는 단계는 각각의 거의 고정된 길이 블록에 대해 실행되고, 도14d의 화살표(1448)와 같은 화살표로 나타나고, 각각의 거의 고정된 길이 블록에 대해 블록 해시를 생성한다. 거의 고정된 길이 블록의 길이와 함께 블록 해시는 제1의 거의 고정된 길이 블록(1440)에 대응하는 블록 기술어(block descriptor, 1450)와 같은 블록 기술어를 포함한다. 블록 기술어의 순열은 대응하는 파일 또는 파일 유사 개체의 거의 고정된 길이 블록에 대해 구성되고, 헤더는 파일 서명(1452)을 구성하기 위한 블록 기술어의 순열에 추가된다. 헤더(1454)는 서명 버전 수를 포함할 수도 있으며, 따라서 파일 서명의 컨텐츠 및/또는 포맷이 시간에 따라 헤더(1454)에 포함된 버전 식별자의 결과로 버전에 관한 각각의 파일 서명 자기 기술을 사용하여 변할 수 있다. 또한, 헤더는 서명 및 추가 정보 내의 블록 기술어의 수를 포함한다. 파일 서명은 헤더 및 블록 기술어를 암호화하는 바이트 시퀀스 또는 바이트 흐름에 의해 나타날 수 있거나, 더욱 복잡한 데이터 구조로 암호화될 수도 있다.

<71> 따라서, 백업-복원 및 보존 시스템의 관점에서, 백업-복원 및 보존 시스템에 의해 계속 감시되고 백업되는 클라이언트 장치내에 저장된 파일 또는 파일 유사 개체의 특정 인스턴스는 파일 서명/데이터 블록 순서쌍으로 간주된다. 파일 또는 파일 유사 개체는 파일 데이터를 포함하는 거의 고정된 길이 블록 및 파일 서명에 의해 충분히 구성되고, 충분히 지정될 수 있다. 클라이언트 내에서, 거의 고정된 길이 블록의 순열은 깨끗한 텍스트 폼이지만 클라이언트로부터 백업 복원 및 보존 시스템의 서버측 부분에 전송된 임의의 거의 고정된 길이 블록이 암호화되어 어떤 외부 엔티티도 그 안에 포함된 데이터에 접근할 수 없을 경우에 사용가능하다.

<72> 도15a 내지 도15e는 본 발명의 실시예에 따르는 파일 인스턴싱을 예시한다. 도15a는 파일의 제1의 기본 레벨 인스턴스를 나타낸다. 전술된 자와 같이, 파일 또는 파일 유사 개체는 클라이언트 장치의 주 서비스 프로세스에 의해 처리되어 백업-복원 및 보존 시스템에 전송하고 백업-복원 및 보존 시스템 내의 저장장치에 전송하기 위해 압축되고 암호화되는 거의 고정된 길이 블록(1504)의 순서 및 서명(1502)을 생성한다. 따라서, 파일 서명/거의 고정된 길이 블록 순서쌍은 파일 또는 파일 유사 개체의 콘텐츠를 충분히 한정하고, 이 콘텐츠로부터 파일 서명 및 거의 고정된 길이 블록이 생성된다. 처음에, 시간에 따라 파일의 변화를 계속해서 백업하고 감시하기 위해 클라이언트 장치 사용자에 의해 파일이 식별된다. 각각의 파일 변화 시간은 검출되고, 파일은 백업되며, 파일의 새로운 인스턴스 또는 버전은 백업-복원 및 보존 시스템의 관점으로부터 생성된다. 먼저 파일이 연속적인 백업에 대해 지정될 경우, 도15a에 나타난 바와 같이 파일 서명/암호화된 거의 고정된 길이 블록 순서쌍이

생성되고, 파일의 기본 레벨 인스턴스를 나타내기 위해 백업-복원 및 보존 시스템의 서버측 부분에 전송된다.

<73> 도15b는 도15a에 나타난 파일 서명/임호화된 거의 고정된 길이 블록 순서쌍에 의해 설명된 파일의 제1 백업을 예시한다. 파일이 발견되어 변경 및 편집되면, 새로운 파일 서명(1506)이 파일의 현재 콘텐츠로부터 생성된다. 새로운 파일 서명은 이전의 파일 서명(1502)과 정렬된 블록이고, 대응 블록은 도15b의 화살표(1508)와 같은 양방향 수평 화살표로 예시된다. 대응하는 거의 고정된 길이 블록에 대해 계산된 블록 해시가 다르면, 대응하는 거의 고정된 길이 블록은 원본 파일에 대해 변경된 것이다. 도15b에서, 블록 기술어(1510-1512)에 의해 표시되는 거의 고정된 길이 블록은 제1 파일 서명(1502) 및 새로운 파일 서명(1506)의 대응하는 블록 기술어의 비교에 의해 변경되었다고 판단된다. 이 파일 기술어(1514-1516)와 관련된 거의 고정된 길이 블록은 모두 수정된 블록 또는 Δ블록의 리스트를 포함한다. 도15b를 참고로 하여 논의된 파일 서명 비교는 지능적인 비교로서, 새로운 블록에 대한 고유 블록 순서, 삭제될 고유 블록 및 수정된 파일의 대응 블록과 파일의 제1 인스턴스의 블록 사이의 일치를 깨지 않고 발생시키기 위해 파일의 대규모 수정 내에 새로운 블록이 삽입되는 것을 허용한다. 바꿔 말하면, 두 개의 파일 서명은 일치되게 배치될 수 있으며 삽입 및 삭제 모두 검출되어 하나의 파일 서명에서 삭제 및/또는 추가 다음에 블록 기술어는 다른 서명의 블록 기술어와 여전히 일치하고, 유전자 자리(gene loci)에 대응하는 매우 유사한 DNA 서열은 서브시퀀스의 삽입 및 삭제에도 불구하고 서로 배치될 수 있다.

<74> 따라서, 도15b에 나타난 바와 같이, 다음과 같은 파일 수정 또는 교체, 새로운 파일 서명 및 Δ블록 셋의 검출이 생성되어 원본 파일 서명(1502) 및 거의 고정된 길이 블록(1504)의 순열의 결합은, 새롭게 생성된 파일 서명(1506) 및 Δ블록(1514-1516)과 함께 원본 파일 및 원본 파일의 교체 또는 수정된 버전인 서브시퀀트 모두를 충분하게 설명한다. 도15c에 나타난 바와 같이, 원본 파일 및 원본 파일의 수정된 버전 모두의 콘텐츠는 파일의 인스턴스 0 및 인스턴스 1로 간주되며, 일반적으로 작은 Δ블록(1518) 셋뿐만 아니라 거의 고정된 길이 데이터 블록의 원래 순열을 포함한다. 원본 블록 및 Δ블록은 백업-복원 및 보존 시스템의 서버측 부분의 영구 스토어 내에 저장되고, 원본 파일 서명(1502) 및 더욱 최근에 생성된 제2의 파일 서명(1506)은 백업-복원 및 보존 시스템의 서버측 부분의 데이터베이스 부분에 저장되며, 원본 파일 또는 그 후에 수정된 파일 중 하나는 저장된 데이터 블록, Δ블록 및 파일 서명으로부터 완전하게 복원될 수 있다.

<75> 도15d에 나타난 바와 같이, 각각의 검출된 수정 파일과 함께, Δ블록의 결정 및 영구 데이터 스토어 내의 Δ블록의 저장장치, 파일의 새로운 인스턴스가 생성된다. 도15d에서, 파일의 6개의 인스턴스가 원본 파일의 백업에 뒤에 생성된다. 언급된 바와 같이, 원본 파일은 거의 고정된 길이 블록의 열로 표시되고, 각각의 후속 인스턴스는 Δ블록(1518, 1523)의 열로 표시된다. 원본 파일 블록 및 각각의 인스턴스에 대한 Δ블록만을 저장함으로써, 각각의 인스턴스 전부를 저장하는 것보다 많은 작은 수의 데이터 블록이 저장되어 파일의 모든 인스턴스를 나타낸다. 게다가, 아래에 더욱 자세하게 언급되는 것처럼, 데이터 블록이 백업-복원 및 보존 시스템의 서버측 부분의 영구 스토어에 저장되고 그들의 개별 블록 해시에 의해서만 색인되기 때문에, 클라이언트 장치의 다수의 파일 또는 다수의 클라이언트 장치에 걸쳐서 분배되는 다수의 파일 내에 나타나는 데이터 블록은 영구 스토어에 오직 한번만 저장되어야 한다. 바꿔 말하면, 백업-복원 및 보존 시스템에 의해 연속적으로 감시되고 백업되는 다양한 클라이언트 장치에 걸쳐 분배되는 다양한 파일 내에 특정 데이터 블록이 얼마나 여러 번 발생하는지와는 상관없이, 특정 블록 해시 값에 의해 식별된 임의의 특정 데이터 블록의 단일 인스턴스만 영구 스토어에 저장될 필요가 있다. 도15d에 나타난 바와 같이, 파일의 모든 상이한 인스턴스를 나타내기 위해 영구 스토어 내에 저장된 원본 파일 블록 및 Δ블록과 함께, 파일의 각각의 연속적인 인스턴스를 위해 생성된 파일 서명의 완전한 셋은 백업-복원 및 보존 시스템의 서버측 부분의 데이터베이스 부분 내에서 유지된다. 본 발명의 일부 실시예에서, 파일 서명은 차등 저장(differential storage) 기법을 사용하여 저장될 수 있고, 차등 저장을 사용하여 정확하게 데이터 블록만 저장된다. 바꿔 말하면, 제1 파일 서명이 영구 스토어 전체에 저장될 수도 있으며, 다음의 인스턴스와 이전에 저장된 인스턴스에 대해 계산된 파일 서명 사이의 차이만 저장된다.

<76> 도16은 백업-복원 및 보존 시스템에 의해 연속적으로 감시되고 백업되는 클라이언트 장치의 각각의 파일을 위해 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템의 클라이언트측 부분 및 서버측 부분에 저장된 정보를 요약한다. 도16에 나타난 바와 같이, 백업-복원 및 보존 시스템의 서버측 부분에서, 파일(1602)에 대한 파일 서명 이력이 도15e에 논리적으로 도시된 바와 같이 백업-복원 및 보존 시스템의 서버측 부분의 데이터베이스 부분 내에 저장된다. 또한, 도15d에 예시된 바와 같이, 파일 데이터 블록 이력(1604)의 압축되고 암호화된 버전은 백업-복원 및 보존 시스템의 서버측 부분의 영구 스토어 부분 내에 저장된다. 클라이언트 측에서는, 파일(1606)에 대해 가장 최근에 생성된 파일 서명이 감시되고 백업된 각 파일에 대해 저장되는 것이 바람직하다. 마지막의 가장 최즌에 생성된 파일 서명의 로컬 카피를 유지하여, 파일의 후속 인스턴스 및 파일의 가장 최근에 저장된 인스턴스 사이의 차이는 백업-복원 및 보존 시스템의 서버측 부분에 저장된 정보에 접근할 필요없이, 저장된

파일 서명(1606) 및 파일의 새로운 인스턴스에 대해 생성된 새로운 파일 서명으로부터 전체적으로 계산될 수 있다. 또한, 클라이언트측 데이터 저장 리소스를 인정하는 것처럼, 서명 캐시(1608) 및 데이터 블록 캐시(1610)가 클라이언트 장치에 유지되어 복원 동작을 촉진시킬 수 있다. 가장 양호한 경우에, 파일은 백업-복원 및 보존 시스템의 서버측 부분으로부터 파일 서명 및 데이터 블록을 검색할 필요없이 오직 국부적으로 저장된 파일 서명 및 데이터 블록을 사용하여 이전의 버전 또는 인스턴스로 복원될 수 있다. 그러나, 서명 캐시(1608), 데이터 블록 캐시(1610) 및 심지어는 가장 최근에 생성된 파일 서명(1606)이 클라이언트 장치로부터 삭제되어야하며, 파일의 임의의 이전에 생성되고 백업된 인스턴스는 백업-복원 및 보존 시스템의 서버측 부분으로부터 필요한 파일 서명 및 데이터 블록에 먼저 접근함으로써 클라이언트 장치에 복원될 수 있다.

<77> 도17a 내지 도17b는 본 발명의 실시예에 따르는 파일에 대해 저장된 파일 서명 이력 및 데이터 블록 이력으로부터 파일의 특정 인스턴스를 구성하기 위한 논리적 동작을 예시한다. 도17a는 도15e를 참고로 하여 이미 언급된 것처럼 파일에 대한 파일 서명 이력을 나타낸다. 파일의 완전한 가장 최근의 인스턴스를 구성하기 위해, 인접한 파일 서명의 블록 해시 사이의 차이가 삭제될 때까지 파일 서명 이력내의 각 블록에 대한 대응 블록 기술어는 차단될(traverse) 필요가 있다. 예를 들어, 파일의 제6 인스턴스의 제1 데이터 블록에 대해, 블록 시술어(1702)로 표시되는, 파일 서명 이력내의 블록에 대응하는 블록 기술어는 두 개의 인접 파일 서명을 검출하기 위해 가장 최근으로부터 최소 최근(least recent)까지 차단된다. 도17a에 나타난 바와 같이, 파일의 제5 인스턴스(1708) 및 제4 인스턴스(1710)에 대응하는 파일 서명의 대응블록 기술어(1704, 1706)에 저장된 블록 해시의 비교는 다르다고 검출되며, 파일의 제5 인스턴스에 대해 계산된 차분 블록(difference block)은 파일의 제6 인스턴스 내에 포함될 데이터 블록의 버전을 나타낸다. 도17b는 도15d를 참고로 하여 앞서 언급된 것처럼, 파일에 대한 데이터 블록 이력을 나타낸다. 도17b에서 볼 수 있듯이, 파일(1712)의 첫번째 데이터 블록에 대응하는 가장 최근에 저장된 데이터 블록은 파일의 제5 인스턴스의 백업동안 검출되고 저장되는 차분 블록이다. 다른 예처럼, 파일의 최종 블록에 대한 모든 블록 해시 및 모든 파일 서명은 동일하고, 원본 파일이 저장되었기 때문에 파일의 최종 블록은 변하지 않는 것을 나타내고, 따라서 원본 데이터 블록은 인스턴스(6)에 포함되어야만 하는 데이터 블록이다.

<78> 도17a 내지 도17b는 파일에 대한 파일 서명 이력 및 데이터 블록 이력으로부터 주어진 논리적 복구를 예시하기 위한 것이다. 그러나, 실용적인 관점으로부터, 파일의 특정 인스턴스는 인스턴스 및 데이터 블록 스토어에 대응하는 파일 서명만을 사용하여 완전하게 복원될 수 있다. 이것은 파일 서명내의 각각의 블록 기술어가 파일 내의 대응 데이터 블록에 발생하는 데이터 블록을 유일하게 설명하는 블록 해시를 포함하기 때문이다.

<79> 도18a 내지 18b는 본 발명의 실시예를 따르는 버전 이력 절단(version history truncation)을 예시한다. 도18a는 도15d를 참고로 하여 논의된 블록 이력을 나타낸다. 그것은 저장 공간을 보존하기 위해, 파일의 가장 최근 인스턴스의 단지 몇 개에만 저장하기 위해 백업-복원 및 보존 시스템이 선택할 수도 있는 경우를 나타낼 수도 있다. 예를 들어, 도18a에 나타난 바와 같이, 백업-복원 및 보존 시스템은 파일에 대해 인스턴스(6,5,4)만을 저장하기 위해 선택할 수도 있고, 인스턴스(2,1,0)에 대해 파일 서명 및 불필요한 블록을 삭제할 수도 있다. 개념적으로, 인스턴스 이력 절단, 또는 최소 최근에 생성된 다수의 인스턴스 제거는 도18a에 점선(1802)으로 표시된 새로운 최소 최근에 생성된 인스턴스를 선택하는 것으로 생각될 수 있고, 이전 인스턴스에 대해 불필요한 데이터 블록뿐만 아니라 이전 인스턴스에 대해 저장된 파일 서명을 제거하는 것으로 생각될 수 있다. 도18a에서, 불필요한 데이터 블록은 "X" 기호(1804)와 같은 "X" 기호로 표시된다. 파일의 세번째 인스턴스의 데이터를 나타내는 데이터 블록은 열린 원(1806)과 같은 열린 원으로 표시된다. 따라서, 파일 이력을 절단하기 위해, 인스턴스(2,1,0)에 대응하는 파일 서명이 제거되고, 도18a에 "X" 기호로 표시된 데이터 블록은 영구 스토어에서 삭제될 수 있다. 도18b는 도18a를 참고로 언급된 버전 절단을 따르는 데이터 블록 이력을 나타낸다. 도15b에 나타난 바와 같이, 이전에 "인스턴스 3"으로 분류된 인스턴스는 "인스턴스 0"(1810)으로 분류되고, 인스턴스 3 이전 또는 다음에 변조된 이전 의 인스턴스로부터 데이터 블록이 제거되었다. 따라서 인스턴스 이력은 원본 파일을 재구성하고, 성공적인 인스턴스를 생성하고, 제거될 가장 최근에 생성된 인스턴스 포함할 필요가 없이 백업-복원 및 보존 시스템의 서버측 부분에서 완전히 절단될 수 있다.

<80> 도19a 내지 도19b는 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템 내의 보안관련 엔티티 및 동작을 예시한다. 이 엔티티 및 동작은 백업-복원 및 보존 시스템의 클라이언트측, 서버측 및 파트너측에 관해 언급되었다. 이 엔티티 및 동작의 설명에는, 단일 클라이언트 장치(1902), 서버측 부분(1904) 및 파트너(1906)가 고려된다. 클라이언트, 파트너 및 서버는 안전한 연결부(1908-1910)를 거쳐 서로 통신한다. 본 발명의 일 실시예에서, 파트너/서버 통신은 이중 인증된 SSL 연결부를 통해 전달된다. 클라이언트/파트너 통신은 클라이언트를 대신하여 제3 파티의 인증 서비스에 의해 인증된 SSL 인증서를 제공하는 파트너를 사용하여 단일측의 SSL 안전

한 연결부(1908)를 통해 전달된다. 초기의 클라이언트/서버 통신 동안, 단독으로 인증된 SSL 연결부(1910)가 사용된다. 이어서, SSL 연결부는 서버로의 클라이언트의 각각의 요청을 서버로의 신임장 전송에 의해 보충된다.

<81> 파트너(1906)는 그것을 통하여 클라이언트가 서비스에 대해 계약하는 서버로부터 독립된 엔티티이다. 파트너도 또한 아래에 언급하는 바와 같이, 전체 보안 전략의 아주 중요한 구성요소이다. 파트너는 백업-복원 및 보존 시스템의 서버측 부분에 파트너 공용키(1914)를 제공하고, 그 결과 서버에 의해 클라이언트로 제공되고 파트너 공용키(1912)를 안전하게 파트너 시스템에 유지하는 파트너 개인키/공용키 암호키 쌍을 생성한다. 파트너는 또한 클라이언트를 식별하는 저장된 장치 ID(1916)를 포함한다. 클라이언트는 또한 장치 ID를 저장한다. 장치 ID는 처음에 서버(1904)에서 생성되고 서버(1904) 내에 저장된다. 서버는 클라이언트를 대신하여 신임장을 생성하고, 그후의 클라이언트/서버 통신을 보장하기 위해 클라이언트에게 신임장을 제공한다. 클라이언트는 오직 클라이언트에게만 알려진 파일 암호키(1920)를 생성하고 사용한다. 파일 암호키는 서버(1904)에 전송되고, 서버(1904)에 의해 저장된 데이터 블록을 암호화하는데 사용된다. 클라이언트는 또한 이중으로 암호화된 형태(1924)의 서버내의 저장장치에 대해 클라이언트의 파일 암호키(1920)를 이중으로 암호화하기 위해 파트너 공용키(1912)와 함께 사용되는 클라이언트 암호키(1922)를 생성하고 저장한다.

<82> 도19b는 파일 암호키 사용을 예시한다. 파일 암호키(1920)는 클라이언트에 의해 사용되어 서버(1904)로 전송되고 서버(1904)에 의해 저장되는 각각의 게이터 블록(1930)을 암호화하는데 사용된다. 비슷하게, 클라이언트는 파일 암호키(1920)를 사용하여 클라이언트 파일 인스턴스를 복원하기 위해 사용되는 클라이언트로 서버에 의해 되돌아간 데이터 블록을 암호화한다. 파일 암호키가 클라이언트에 의해 생성되고 클라이언트에만 접근가능하기 때문에, 백업 및 복원 동작의 일부로써 클라이언트 장치로부터 원격의 엔티티로 전송된 어떤 파일 데이터도 원격 장치에 의해 접근할 수 없다. 비록 파일 암호키가 서버(1904) 내에서 날인되어도, 파일 암호키는 클라이언트에게만 알려진 클라이언트 암호키(1922) 및 파트너 공용 암호키(1912) 모두에 의해 자신이 암호화되기 때문에, 서버는 파일 암호키에 접근할 수 없다. 클라이언트는 파트너가 클라이언트 암호키에 의해 암호화된 파일 암호키를 클라이언트 장치에 돌려보내고, 암호의 제1층을 해독하고 서버상의 날인된 이중으로 암호화된 파일 암호키를 검색하려는 요구에 의해 파일 암호키를 회복할 수도 있다. 이것은 서버가 클라이언트 데이터 또는 클라이언트의 파일 암호키에 접근할 수 없고, 파트너가 파일 암호키 또는 클라이언트 데이터 모두에 접근할 수 없다는 것을 보장한다. 파일 암호키는 서버(1904) 내에서 날인되기 때문에, 클라이언트에 의한 파일 암호키의 손실은 클라이언트에게 치명적이다.

<83> 도19c는 파일 암호키를 무심코 삭제하거나 분실하는 경우 클라이언트 장치에 의한 파일 암호키 복구를 예시한다. 파일 암호키 없으면, 클라이언트는 서버에 의해 클라이언트로 돌아온 암호화된 데이터 블록을 해독할 수가 없다. 그러나, 클라이언트 파일 암호키는 서버(1904)상에 이중으로 암호화된 형태(1924)로 날인된다. 따라서, 파일 암호키를 복구하기 위해, 클라이언트는 파일 암호키를 복구하려는 요청(1936)을 파트너(1906)에게 발송하고, 그 결과, 파트너는 요청을 서버(1904)에게 전송한다. 서버는 이중으로 암호화된 파일 암호키(1924)를 파트너에게 돌려보내고, 파트너는 제1 단계의 암호를 해독하여 클라이언트의 클라이언트 암호키(1938)와 함께 단독으로 암호화된 파일 암호키를 생성한다. 단독으로 암호화된 파일 암호키(1938)는 클라이언트(1902)에게 돌려 보내지고, 클라이언트는 클라이언트 암호키를 사용하여 파일 암호키를 해독하여 깨끗한 형태(1940)의 파일 암호키를 재생성한다.

<84> 도19d는 클라이언트 신임장에 의해 클라이언트 장치 및 서버 사이의 안전한 통신을 예시한다. 서버가 서버로부터 현재 변조된 클라이언트 파일을 백업하기 위한 업로드 파일의 처리와 같은 서비스를 요청할 경우, 클라이언트는 요청내에 신임장(1918)과 함께 클라이언트를 나타내는, 서버에 의해 생성되고 클라이언트 초기화동안 클라이언트 컴퓨터에 공급되는 사용자 이름 및 패스워드와 같은 장치 ID(1916)를 포함한다. 요청을 수신하면, 백업-복원 및 보존 시스템의 서버측 부분의 활성 디렉토리 부분내의 신임장 및 저장된 장치 ID(1916)을 구비한 요청과 함께 포함된 장치 ID와 신임장을 매칭시켜, 유효한 클라이언트로부터 요청이 수신되었다는 것을 서버가 확인할 수 있다. 따라서, 장치 ID 및 신임장은 서버가 개별 클라이언트로부터 수신한 바와 같은 요청을 식별하도록 하고, 백업-복원 및 보존 시스템의 서버측 부분을 통해 요청을 발송하여, 요청으로부터 생성된 응답은 요청된 클라이언트로 돌아갈 수 있다.

<85> 도20a 내지 도20c는 클라이언트의 초기화를 예시하는 제어 흐름도의 한 형태를 제공하며, 클라이언트는 충분히 안전한 요청 및 데이터 교환을 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템의 서버측 부분에 전달할 수 있다. 먼저 단계 2002에서, 클라이언트는 백업 및 복원 서비스를 수신하기 위한 요청을 준비하고 파트너에게 발송한다. 클라이언트는 요청을 준비하고 파트너 제공 웹 페이지, 파트너 제공 초기화 경로 또는 다른 방법을 거쳐서 발송할 수도 있다. 단계 2004에서, 파트너는 클라이언트로부터 백업 및 복원 서비스 요청을 수신

하고, 단계 2006에서, 합리적인 클라이언트 장치로부터 나온 요청을 확인하고, 클라이언트 장치와 호감이 가지 않는 장치를 부합시키기 위한 시도와 같은 임의의 다른 추가 확인을 실행하고, 클라이언트에 대해 저장된 데이터 엔트리를 정하여, 파트너는 결국 클라이언트를 추적하고 이어서 클라이언트를 식별하고 클라이언트와 상호작용하고, 장치 공급 요청을 서버에 발송함으로써 서버로부터 새로운 장치의 공급을 요청할 수도 있다. 단계 2008에서, 수신기는 장치 공급 요청을 수신하고, 단계 2010에서 일회용 장치 티켓을 준비한 다음 서버가 파트너에게 돌려보낸다. 일회용 장치 티켓은 URL을 포함함으로써 클라이언트가 이어서 연속적으로 서버(2012), 백업-복원 및 보존 시스템(2014) 내의 일회용 티켓을 유일하게 식별하는 티켓 ID 및 클라이언트(2016)를 나타내기 위해 생성되는 장치 ID에 접촉한다. 단계 2018에서, 파트너는 일회용 장치 티켓을 수신하고 파트너에 의해 저장된 클라이언트 정보를 업데이트하여 단계 2020에서 티켓을 클라이언트 장치로 전송하기 전에 티켓 안에 포함된 장치 ID를 포함한다. 또한, 파트너는 티켓을 클라이언트 장치로 전송하기 전에 선택가능한 정보를 티켓으로 발송할 수도 있다. 예를 들어, 클라이언트 장치 그룹이 선택되어 파트너에 의해 클라이언트를 대신하여 생성된 다른 정보 및 일반적인 클라이언트 암호 키를 사용할 수도 있다. 티켓에 추가된 선택가능한 정보는 이 키를 포함할 수도 있다. 단계 2022에서, 클라이언트 장치는 일회용 티켓을 수신한다. 단계 2024에서, 클라이언트는 도13a를 참고로 하여 언급된 클라이언트측 사용자 인터페이스 경로 및 서비스 프로세스를 구현하는 클라이언트측 애플리케이션을 설치한다. 클라이언트측 의 실행가능한 것이 설치되고 실행되면, 단계 2026에서, 클라이언트는 서버와 안전한 연결부를 만들고 클라이언트에 의해 수신한 일회용 장치 티켓에 포함된 티켓 ID(2014)를 서버로 발송한다. 단계 2028에서, 서버는 티켓 ID를 수신한 다음, 도10b에 계속해서, 티켓 ID를 허가하고, 단계 2030에서 클라이언트에 대한 파트너 정보 및 이전에 생성된 장치 ID를 인정한다. 그 다음 서버는 백업-복원 및 보존 시스템의 서버측 부분을 구성하여 단계 2032에서 장치 ID에 의해 식별된 클라이언트 장치로 서비스를 제공하고, 단계 2034에서는 패스워드 및 사용자 이름과 같은 신임장을 클라이언트 장치에 대해 생성한다. 그 다음에 단계 2036에서, 서버는 생성된 사용자 이름(2040), 패스워드(2042) 및 파트너의 공용키(2044)를 포함하는 클라이언트(2038)에 대한 응답을 준비하고, 응답(2038)을 클라이언트에게 돌려보낸다. 단계 2050에서, 클라이언트는 응답을 수신하고, 그 다음, 단계(2052)에서, 응답에 포함된 정보를 사용하여, 연속되는 요청 및 서버와의 데이터 교환을 위해 클라이언트 측 프로세스를 구성한다. 단계 2054에서, 클라이언트는 클라이언트의 파일 암호키를 생성하고 패스워드 기반 방법을 거쳐 클라이언트의 클라이언트 암호키를 생성한다. 패스워드를 기억함으로써, 클라이언트는 연속하여 클라이언트 암호키를 재생성할 수 있다. 그 다음에, 단계 2056에서, 클라이언트는 파일 암호키를 클라이언트 암호키로 암호화하고, 그 다음에 암호화된 파일 암호키를 파트너의 공용키로 암호화하여 이중으로 암호화된 파일 암호키를 생성한다. 단계 2058에서, 클라이언트는 이중으로 암호화된 파일 암호키를 서버에게 발송하고, 서버는 단계 2060에서 이중으로 암호화된 파일 암호키를 수신하고, 저장하거나, 이중으로 암호화된 파일 암호키를 날인한다. 단계 2062에서, 클라이언트는 클라이언트에게 인지(acknowledgement)를 돌려보내며, 단계 2064에서 인지를 수신할 때 서버에게 요청을 연속적으로 발행하고 서버와 데이터를 교환하기 위해 준비된다. 클라이언트 및 서버 사이의 안전한 연결부는 짧은 시간 동안만 동작할 수 있으며, 클라이언트에 대한 연속적인 요청 및 데이터 교환을 위해 재구성될 수도 있다. 클라이언트가 장치 ID 및 신임장을 소유하면, 클라이언트는 어떤 시간에도 서버에 충분히 안전한 연결부를 재구성할 수 있다.

<86> 도21은 전체 레벨에서, 본 발명의 일 실시예를 나타내는 백업-복원 및 보존 시스템의 서버측 부분의 영구 스토어 부분에 의해 구현된 블록 스토어를 예시한다. 도21에서, 블록 스토어(2102)는 블록 해시 인덱스(2104)를 포함하는 것처럼 예시된다. 블록 해시 인덱스의 각각의 엔트리는 블록 스토어 내에 저장된 데이터 블록(2106)과 같은 특정의 암호화된 데이터 블록을 참고한다. 특정 데이터 블록에 대한 참고를 포함하는 것 이외에, 블록 해시 인덱스의 엔트리는 현재 블록을 참고로 하는 파일 서명의 수를 표시하기 위해 참고 횟수를 포함할 수도 있다. 이렇게 하여, 다수의 클라이언트에 걸쳐 분배된 다수의 파일이 데이터 블록을 포함할 수도 있다는 사실에도 불구하고, 임의의 특정 데이터 블록의 단일 인스턴스만이 데이터 스토어에 저장될 필요가 있다. 블록 스토어에 의해 제공된 동작은 (1) 질문;(2) 회복; (3) 저장; 및 (4) 삭제를 포함한다. 질문 동작(2018)에서, 블록 스토어는 블록 해시(2110)을 수신하고 블록 해시 인덱스를 참고로 하여 블록 해시에 대응하는 암호화된 데이터 블록이 블록 스토어에 현재 저장되었는지 아닌지를 판단한다. 블록 해시에 대응하는 데이터 블록이 현재 블록 스토어에 저장되어 있는지 아닌지의 표시(2112)가 돌려 보내진다. 회복 동작(2114)에서, 블록 스토어는 블록 해시(2116)를 수신하고 현재 저장된 암호화된 데이터 블록이 공급된 블록 해시(2116)에 대응할 경우 블록 스토어로부터 블록 해시에 대응하는 암호화된 데이터 블록(2118)을 돌려보낸다. 저장 동작(2120)에서는, 블록 스토어는 블록 해시, 암호화된 데이터 블록을 수신하고, 암호화된 데이터 블록이 블록 스토어 내에 이미 저장되지 않았을 경우, 데이터 블록을 저장하고 데이터 블록을 참고하기 위해 블록 해시 인덱스를 공급된 블록 해시에 업데이트한다. 암호화된 데이터 블록이 블록 스토어 내에 이미 저장되어 있을 경우, 데이터 블록에 대한 참고 횟수는 증

가한다. 삭제 동작(2126)에서, 블록 해시가 블록 해시 인덱스 내에 현재 존재할 경우, 블록 스토어는 블록 해시 및 블록 해시에 대한 참고 횟수의 감소를 수신한다. 참고 횟수가 0까지 감소하면, 블록 해시에 대응하는 블록 해시 인덱스 엔트리를 제거하기 전에 블록 해시 인덱스에 의해 참고된 데이터 블록도 또한 삭제된다.

<87> 도21을 참고로 하여 기재된 블록 해시 방법론 및 블록 스토어는 차등 백업 및 복원을 허용한다. 도22는 차등 백업을 예시한다. 도22에 나타난 바와 같이, 새롭게 백업된 파일에 대해 생성된 최종 서명(2202)은, 클라이언트의, 가장 최근의 파일과 파일에 대해 이전에 생성된 파일 서명(2204)을 비교하여 파일의 어느 블록이 변경되었는지(2206)를 판단한다. 이들 변화된 블록에 대한 블록 해시는 서버에 발송될 수 있는 메시지(2208)로 함께 패키지된다. 그 다음에 서버는 도21을 참고로 논의된 바와 같이, 블록 스토어에 질문하여 어느 블록 해시가 블록 스토어의 블록 해시 인덱스에 현재 저장되지 않았는지를 판단한다. 이들 블록 해시에 대응하는 데이터 블록만이 백업 프로세스 동안 클라이언트에 의해 서버로 발송될 필요가 있다. 따라서 서버는 블록 해시 내에 현재 저장되지 않았다는 Δ블록(2206)의 블록 해시 표시(2212)를 보내고, 업로드 파일을 준비하는 클라이언트는 새롭게 생성된 파일 서명(2202) 및 서버에 의해 돌려보내진 블록 해시(2212)에 대응하는 이들 데이터 블록만을 발송할 필요가 있다. 차등 백업은 클라이언트 및 서버 사이의 불필요한 데이터 교환을 제거한다. 두 단계의 커밋 프로토콜(two phase commit protocol)은 데이터 블록이 질문 및 데이터 블록 전송 사이 간격의 데이터 블록 스토어로부터 삭제되지 않았다는 것을 보장하도록 사용될 수 있다.

<88> 도23은 차등 복원을 예시한다. 차등 복원에서, 소정의 인스턴스에 대한 파일 서명(2302)은 클라이언트에 현재 존재라는 파일에 대한 파일 서명(2304)과 비교된다. 비교는 소정의 버전 또는 인스턴스로 파일을 복원하기 위해 회복될 필요가 있는 데이터를 나타내는 Δ블록 셋을 생성한다. 그러나, 이 Δ블록의 일부는 클라이언트에 의해 유지되는 로컬 블록 캐시(2308) 내에 존재할 수도 있기 때문에, Δ블록(2310) 만이 파일을 복원하기 위해 서버로부터 회복될 필요를 소정의 인스턴스에 국부적으로 저장하지 않았다. 로컬 데이터 블록 해시가 광범위하고, 소정의 인스턴스가 상대적으로 최근에 백업된 경우, 복원 동작이 서버로부터 암호화된 데이터 블록을 획득할 필요 없이, 클라이언트 장치에 국부적으로 충분하게 실행될 수도 있다.

<89> 도24a 내지 24b는 본 발명의 일 실시예를 나타내는 주 서비스 프로세스에 의해 실행된 백업 프로세스에 대한 흐름도를 제공한다. 단계 2402에서, 새로운 목록 및 업로드 및 새로운 업로드 파일이 생성된다. 그 다음에, 단계 2404 내지 2411을 포함하는 포-루프(for-loop)에서, 차등 백업에 대해 플래그된 파일 셋의 각각의 파일이 처리된다. 단계 2405에서, 각각의 파일에 대해, 현재 파일 서명이 계산된다. 이전에 계산된 파일 서명이 국부적으로 저장되지 않을 경우, 단계 2406에서 결정된 바와 같이, 단계 2407에서, 이전에 계산된 파일 서명이 서버로부터 복원된다. 다음으로, 단계 2408에서, 파일 서명은 백업 프로세스의 일부로서 서버에 전달되고 암호화될 필요가 있을 수도 있는 Δ블록 리스트를 생성하도록 비교된다. 단계 2409에서, 이 블록 리스트는 업로드 파일로 패키지되고, 차이 비교의 결과는 단계 2410에서 목록에 기록된다. 단계 2411에서 결정된 바와 같이, 더 많은 파일이 처리될 필요가 있을 경우, 제어는 단계 2405로 되돌아 간다. 일단 모든 파일이 처리되면, 단계 2412에서 업로드 파일이 서버로 발송된다. 단계 2414에서, 백업 경로는 서버에 전달될 필요가 실제로 있는 데이터 블록 리스트를 서버로부터 수신한다. 즉, 도22를 참고로 하여 언급된 바와 같이, 서버는 블록 스토어에 질문하여 Δ블록의 어느 것이 블록 스토어에 이미 저장되어 있는지를 결정한다. 계속해서 도24b에서, 단계 2416에서, 백업 경로는 최종 업로드 파일을 열고, 포-루프 비교 단계 2418-2420에서, 파일 서명 및 서버에 저장되고 전송될 필요가 있는 암호화된 데이터 블록은 최종 업로드 파일에 추가된다. 단계 2422에서, 최종 업로드 파일은 서버에 전송되고, 서버는 파일 서명을 데이터베이스, 데이터 스토어의 데이터 블록에 저장하고, 카탈로그 동조 응답(catalog sync response)을 클라이언트에게 돌려보내서 클라이언트가 로컬 카탈로그와 성공적으로 백업된 모든 파일의 파일 서명 이력 업데이트를 포함하는 원격 카탈로그를 동기화시킬 수 있다. 단계 2426에서, 클라이언트는 카탈로그 동조 응답을 수신하고 따라서 로컬 카탈로그를 단계 2428에서 동기화한다. 단계 2430에서, 백업 경로는 카탈로그 동조를 목록과 비교한다. 문제가 서버에 의해 발생한 것으로 드러나고, 단계 2432에서 판단된 바와 같이, 특정 백업 동작이 성공적으로 실행되면, 이러한 문제들은 단계 2434에서 다양한 방법으로 처리된다. 예를 들어, 백업 요청이 재발행되고, 파일은 일시적으로 불안전한 것으로 간주되고, 좀더 포괄적인 복구 기술을 통해 복원될 수도 있으며, 또는 다른 방법이 임의의 당면한 문제를 정정하거나 개선하기 위해 사용될 수도 있다. 마지막으로, 단계 2436에서, 목록이 닫히고 일시적인 파일 및 데이터 구조가 제거되어 백업 동작을 완료한다.

<90> 도25는 본 발명의 일 실시예를 따라 클라이언트 장치에서 실행되는 주 서비스 프로세스에 의해 실행된 복원 동작을 예시하는 제어 흐름도이다. 단계 2502에서, 복원 동작은 파일의 복원을 위해 특정 버전에 실시된다. 단계 2504에서 판단된 바와 같이, 소정의 버전을 위한 파일 서명이 국부적으로 사용가능하지 않을 경우, 단계 2506에서, 파일 서명은 서버로부터 요청된다. 다음으로, 단계 2508에서, 도23을 참고로 논의된 바와 같이, 복원 경로

는 어느 블록이 인스턴스를 복원하기 위해 서버로부터 회복될 필요가 있는지를 판단한다. 단계 2510에서, 블록 필요 리스트는 업로드 파일로 패키지된 다음 단계 2512에서 서버로 전송된다. 단계 2514에서, 복원 경로는 서버로부터 필요한 블록 및 카탈로그 동조 정보를 수신한다. 단계 2516에서, 로컬 카탈로그는 로컬 카탈로그를 업데이트함으로써 동기화되며, 필요할 경우, 파일의 성공적인 복원을 초래한다. 단계 2518에서, 파일에 대한 데이터 블록은 파일의 소정의 인스턴스를 구성하기 위해 조립된 다음, 단계 2520에서, 기존의 파일을 복원된 파일과 교체하기 위해 사용된다. 바꿔 말하면, 도25에 예시된 복원 동작은 기존 파일을 소정의 버전으로 겹쳐 쓴다. 대신에, 복원 동작은 파일의 새로운 인스턴스로, 또는 다른 파일 이름을 갖는 다른 파일로 특정 버전의 파일을 복원하는 것에 관한 것일 수도 있다.

<91> 본 발명의 백업-복원 및 보존 시스템은 특정 암호화 알고리즘, 압축 알고리즘 및 클라이언트 장치에 의해 사용된 특정 파일-암호 키에 대해 융통적이다. 앞서 논의된 바와 같이, 압축 알고리즘, 암호화 알고리즘에 대한 식별자 및 파일 암호키는 블록 해시 계산에 포함되어, 이전의 파일 암호키를 사용하여 파일이 백업된 이후에 클라이언트가 어느 시간에 파일 암호키를 변경할지를 결심할 경우, 클라이언트는 새롭게 생성된 파일 암호키의 사용을 시작할 수 있으며, 서버는 새로운 파일 암호키에 의해 암호화된 데이터 블록의 수신을 시작할 수 있다.시간에 따라, 새로운 파일 암호키를 사용하여 재암호화하고 서버에 재전송하기 위해 서버는 오래된 암호키에 의해 암호화된 데이터 블록을 클라이언트에게 돌려보낸다. 바꿔 말하면, 시간의 특정 기간 동안, 오래된 파일 암호키와 새로운 파일 암호키 모두로 암호화된 데이터 블록은 새로운 파일 암호키로의 이송이 실행되는 것과 동시에 백업-복원 및 보존 시스템에 의해, 모호함이 없이, 유지될 수 있다.

<92> 비록 본 발명이 특정 실시예에 대해서 기재되었어도, 본 발명을 이 실시예에 한정하기 위한 것이 아니다. 본 발명의 사상 이내에서의 수정은 당업자도 알 수 있을 것이다. 예를 들어, 거의 한계가 없는 상이한 웹-서비스 기반 데이터 백업 및 데이터 보존 애플리케이션의 수가 가능하고, 제어 구조, 프로그래밍 언어, 데이터 구조, 모듈화, 및 다른 이와 같은 프로그래밍 파라미터의 전체 호스트를 포함한다. 기재된 실시예가 웹 서비스 플랫폼 및 인터넷 통신을 사용하여 원격의 데이터 백업 및 데이터 보존 서비스를 구현하는 것과 동시에, 상이한 프로토콜 표준 및 사양 및 상이한 통신 매체를 이용하는 본 발명의 실시예를 나타내는 다른 원격 데이터 백업 및 데이터 보존 서비스가 또한 가능하다. 기재된 실시예가 상대적으로 간결한 애플리케이션 인터페이스를 클라이언트 측 및 파트너 서비스 제공자 애플리케이션에 제공하더라도, 다른 실시예는 훨씬 복잡하고 특징이 많은 인터페이스를 제공할 수도 있다. 폭넓은 상이한 공용/개별 암호 체계, 해시 기반 암호화, 대칭적인 암호화, 또는 다른 암호화 기술 중 어떤 것도 본 발명의 다양한 실시예의 데이터 전달 및 클라이언트 초기화를 위해 사용되는 데이터 및 메시지를 암호화하기 위해 이용될 수도 있다. 기재된 실시예가 데이터 파일의 백업 및 보존을 우선 포함하는 것과 동시에, 클라이언트 컴퓨터에 의해 백업되거나 보존되도록 요구되는 데이터 개체의 임의의 형태가 전송 및 데이터 볼트내 저장을 위해 파일 애에 패키지될 수도 있다. 각각의 클라이언트 컴퓨터는 하나 이상의 데이터 볼트에 의해 구성되는 다중 데이터 백업 및 데이터 보존 장치에 관한 것일 수도 있다.

<93> 설명을 목적으로 하는 상기 기재내용은 본 발명의 완전한 이해를 제공하기 위해 특정 체계를 사용했다. 그러나, 당업자는 특정 상세가 본 발명을 실행하기 위해 필요한 것은 아니라는 것을 알 수 있을 것이다. 본 발명의 특정 실시예의 상기 설명은 예시 및 설명을 목적으로 제시된 것이며, 개시된 정확한 형태에 본 발명을 제한하거나 철처하기 규명하기 위한 것이 아니다. 명백하게 많은 변형 및 변화가 상기 개시의 관점에서 가증하다. 실시예는 본 발명의 원칙을 가장 잘 설명하기 위해 나타나고 설명되었으며, 실제 애플리케이션은 다른 당업자가 본 발명을 가장 잘 활용하고, 다양한 변형을 갖는 다양한 실시예는 예상되는 개별적인 사용에 적합하도록 한다.

도면의 간단한 설명

<6> 도1은 소기업의 데이터 백업 및 데이터 저장에 대한 옵션을 예시하는 도.

<7> 도2는 가정 또는 소기업 환경 모두의 PC 사용자에게 이용할 수 있는 추가 수단을 나타내는 도.

<8> 도3은 도1의 도면 번호를 사용하여, 가정의 사용자, 소기업 사용자, 또는 PC 또는 다른 소형 컴퓨터 시스템 사용자에 사용가능한 백업 및 보존 수단을 예시하는 도.

<9> 도4는 원격 데이터 저장장치상에 운용되는 데이터 저장소(data vault) 웹서비스 기반 애플리케이션의 일 실시예에 의해 제공된 2개의 웹서비스 인터페이스를 예시하는 도.

<10> 도5는 본 발명의 다양한 실시예를 지지하는 하나의 가능한 고 레벨 하드웨어 구성을 나타내는 도.

<11> 도6a 내지 6f는 본 발명의 일 실시예를 나타내는 웹서비스 기반 데이터 백업 및 데이터 보존 서비스의 일 양태

를 예시하는 도.

<12> 도7a 내지 도7b는 초기에 클라이언트를 구성하는 것뿐만 아니라 웹서비스 기반 데이터 백업 및 데이터 보존 서비스용 접속을 위해 계약을 하기 위한 클라이언트과 파트너 서비스 제공자 사이의 상호작용을 예시하는 도.

<13> 도8은 전술된 도6a 내지 도6f에 관하여, 데이터 저장소 내에 안전하게 데이터를 저장하기 위해 실시된 클라이언트측 동작을 예시하는 단순한 흐름 제어 프로그램.

<14> 도9는 클라이언트를 위하여 파일을 저장하는 것과 관련된 데이터 저장소 애플리케이션에 의해 수행되는 동작을 예시하는 도.

<15> 도10은 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템의 서버 측 및 클라이언트 측 부분의 단일 서버 구현을 전체 레벨에서 예시하는 도.

<16> 도11은 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템의 서버 측 부분의 단일 서버 구현을 전체 레벨에서 예시하는 도.

<17> 도12는 본 발명의 다른 실시예를 나타내는 복잡하고 중복된 백업, 복원 및 보존 시스템을 예시하는 도.

<18> 도13a 내지 13c는 도10의 전체 레벨에서 예시된 백업, 복원 및 보존 시스템내의 기본 기능성을 예시한 도.

<19> 도14a 내지 14d는 해당 파일 서명 및 암호화된 데이터 블록을 생성하기 위해 유사 파일 개체 및 파일의 주 서비스 공정(도13a의 1314)에 의한 공정을 예시하는 도.

<20> 도15a 내지 도15e는 본 발명의 실시예에 따른 파일 인스턴싱(instancing)을 예시하는 도.

<21> 도16은 백업, 복원 및 보존 시스템에 계속해서 백업되고 감시되는 클라이언트 장치의 각 파일에 대해 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템의 클라이언트 측 부분 및 서버측 부분에 저장된 정보를 요약하는 도,

<22> 도17a 지 도17b는 본 발명의 실시예에 따르는 파일에 대해 저장된 파일 서명 이력 및 데이터 블록 이력으로부터 파일의 특정 인스턴스를 구성하기 위한 논리적 동작을 예시하는 도.

<23> 도18a 내지 도18b는 본 발명의 실시예에 따르는 버전 이력 끊기를 예시하는 도.

<24> 도19a 내지 도19b는 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템내의 보안 관련 엔티티 및 동작을 예시하는 도.

<25> 도19c는 클라이언트가 파일 암호화 키를 분실하거나 우연히 삭제하는 경우 클라이언트 장치에 의한 파일 암호화 키 검색을 예시하는 도.

<26> 도19d는 클라이언트 증명서에 의해 촉진된 클라이언트 장치 및 서버 사이의 안전한 통신을 예시하는 도.

<27> 도20a 내지 도20c는 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템의 서버 측 부분과 데이터를 교환하고 충분히 안전한 요청을 처리할 수 있도록 클라이언트의 초기화를 예시하는 제어 흐름도.

<28> 도21은 본 발명의 일 실시예를 나타내는 백업, 복원 및 보존 시스템의 서버측 부분의 영구 저장부에 의해 구현된 블록 스토어를 전체 레벨에서 예시한 도.

<29> 도22는 차등 백업을 예시하는 도.

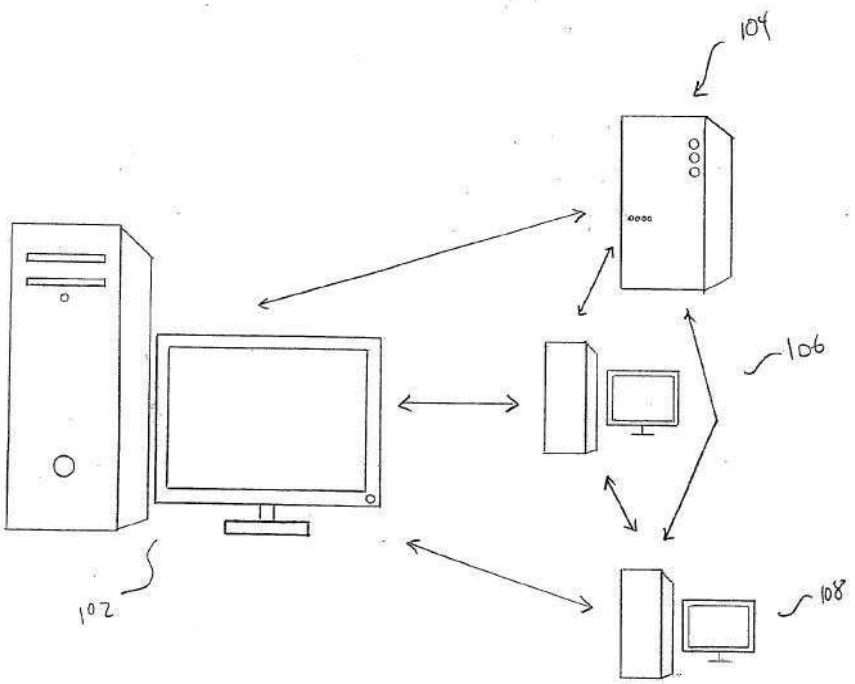
<30> 도23은 차등 복원을 예시하는 도.

<31> 도24a 내지 도24b는 본 발명의 일 실시예를 나타내는 클라이언트측의 백업, 복원 및 보존 시스템의 주 서비스 공정에 의해 실행된 백업 공정에 대한 흐름도.

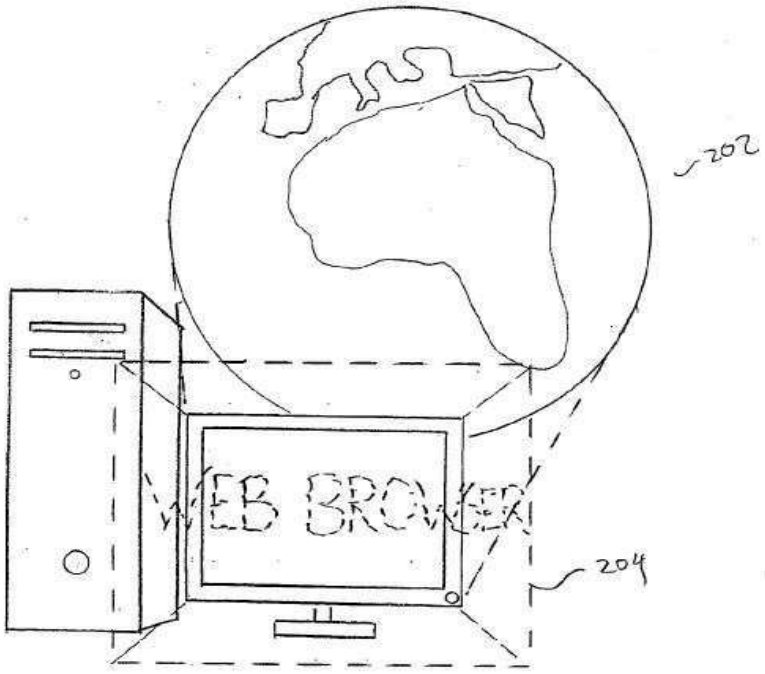
<32> 도25는 본 발명의 일 실시예에 따르는 클라이언트장치에서 실행되는 주 서비스 공정에 의해 실행된 복원 동작을 예시하는 제어흐름도.

도면

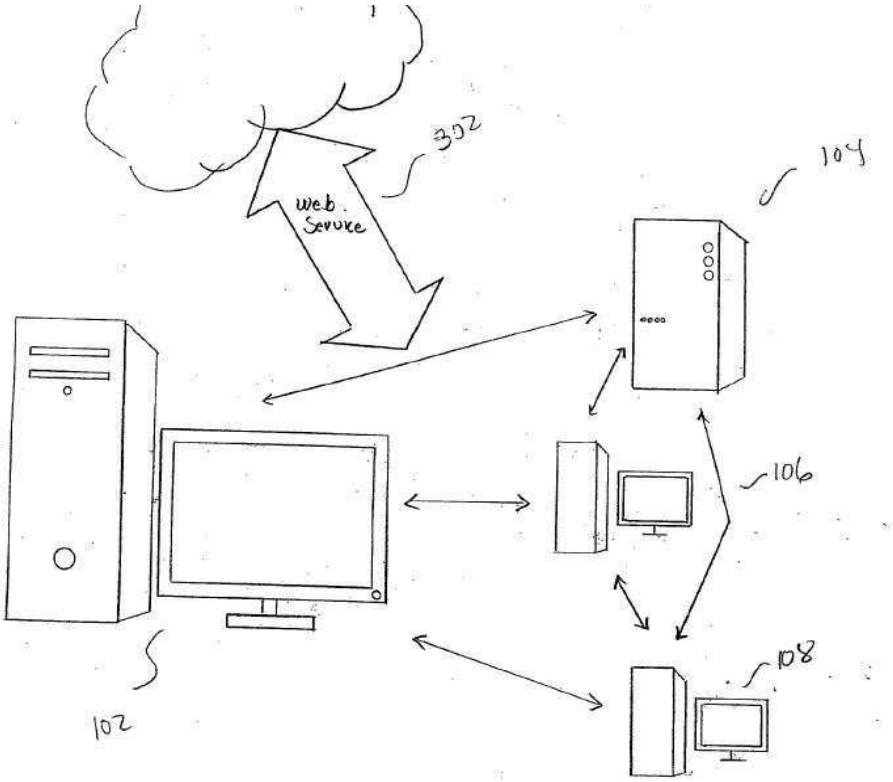
도면1



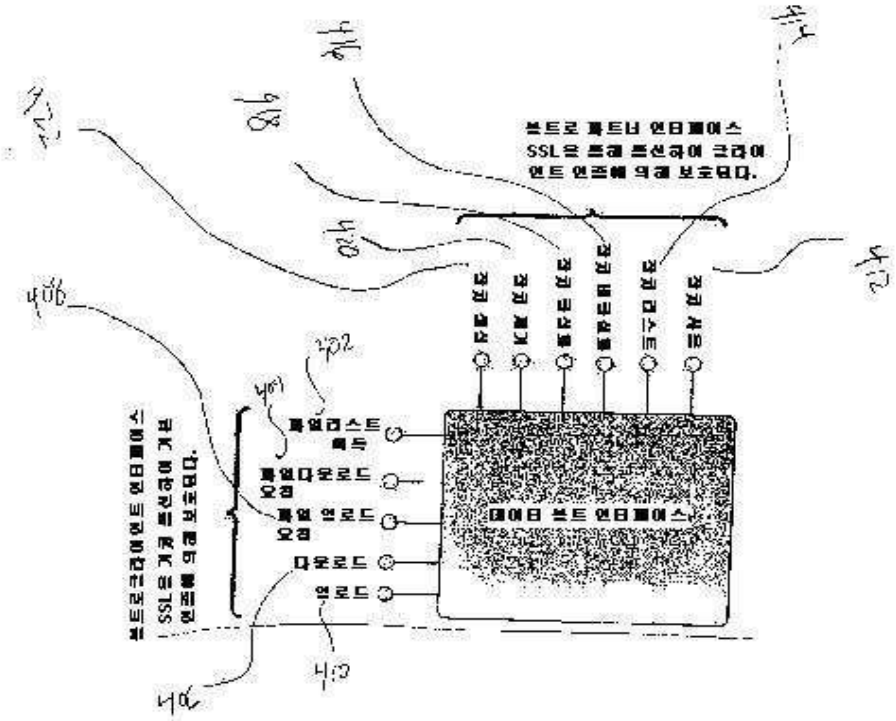
도면2



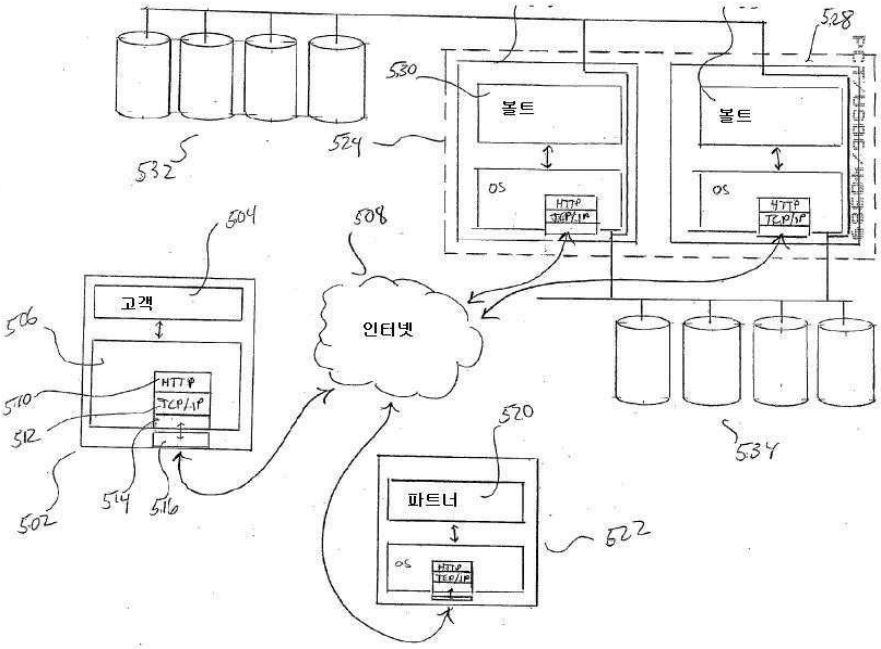
도면3



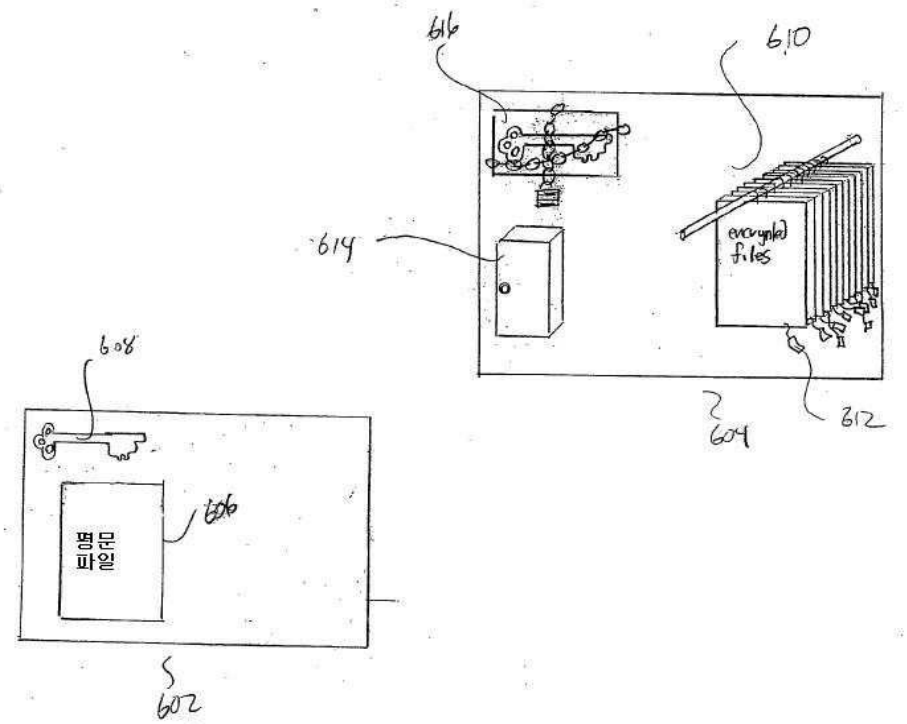
도면4



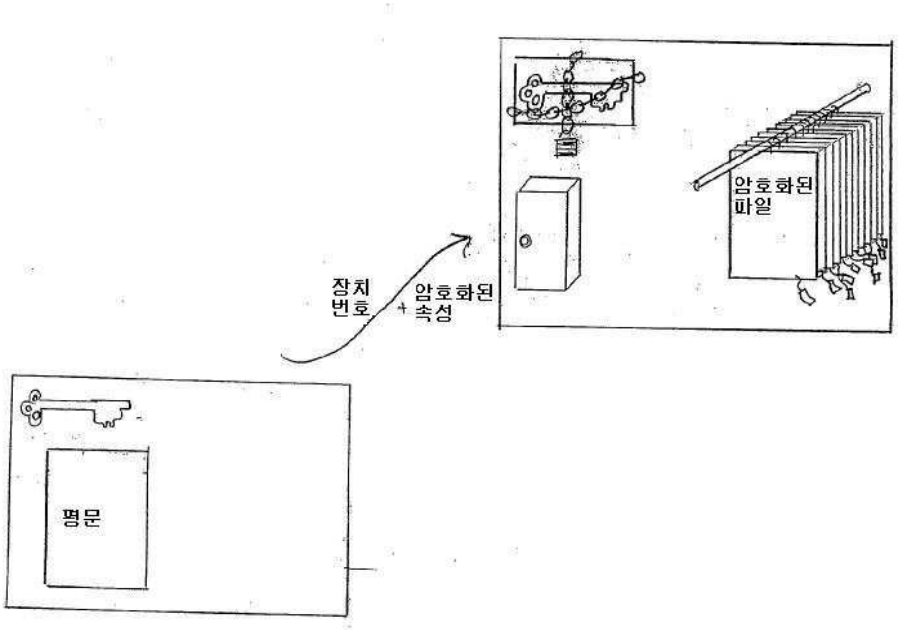
도면5


528
530
532
524
볼트
볼트
OS
OS
HTTP
TCP/IP
504
508
506
고객
인터넷
510
512
502
514
516
520
파트너
522
534

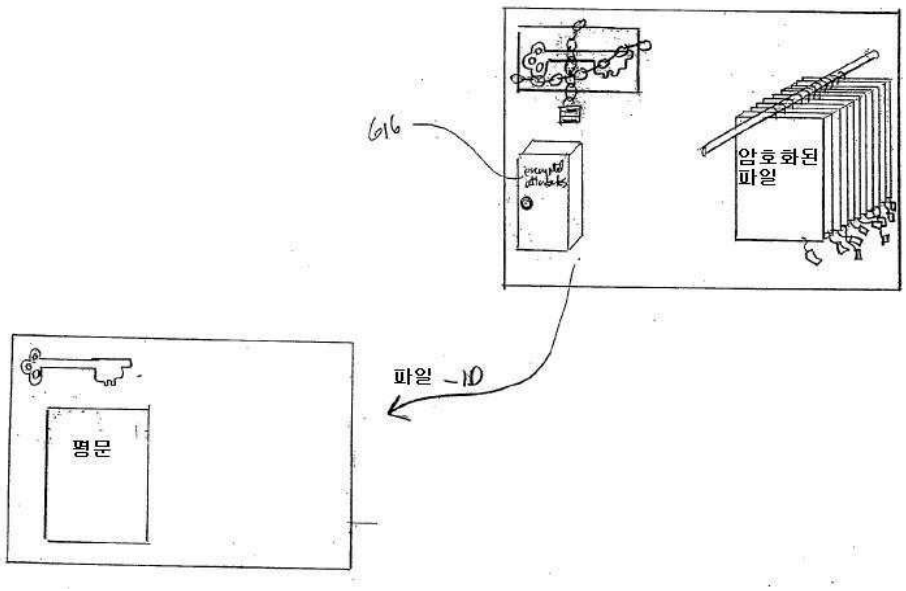
도면6a


616
610
614
encrypted files
612
604
608
606
평문 파일
602

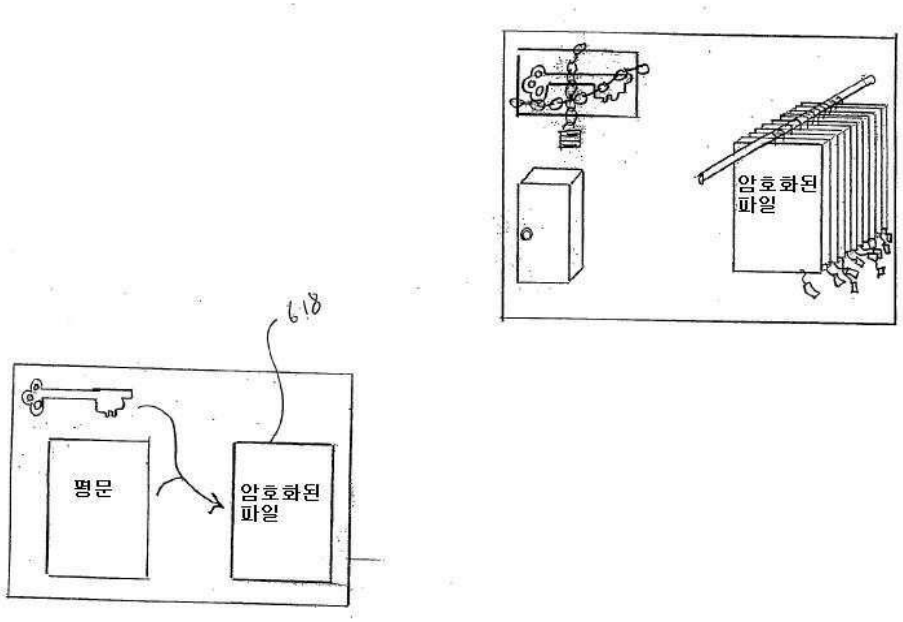
도면6b



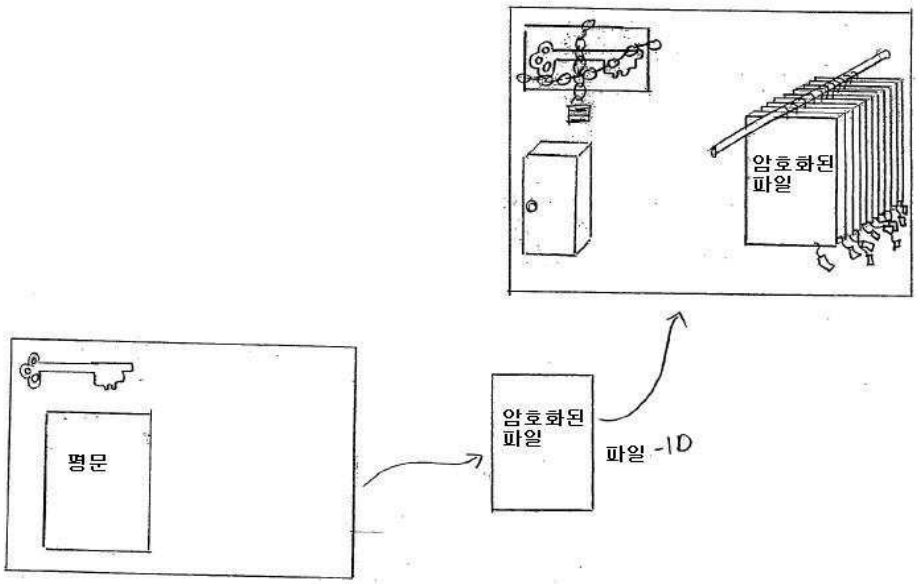
도면6c



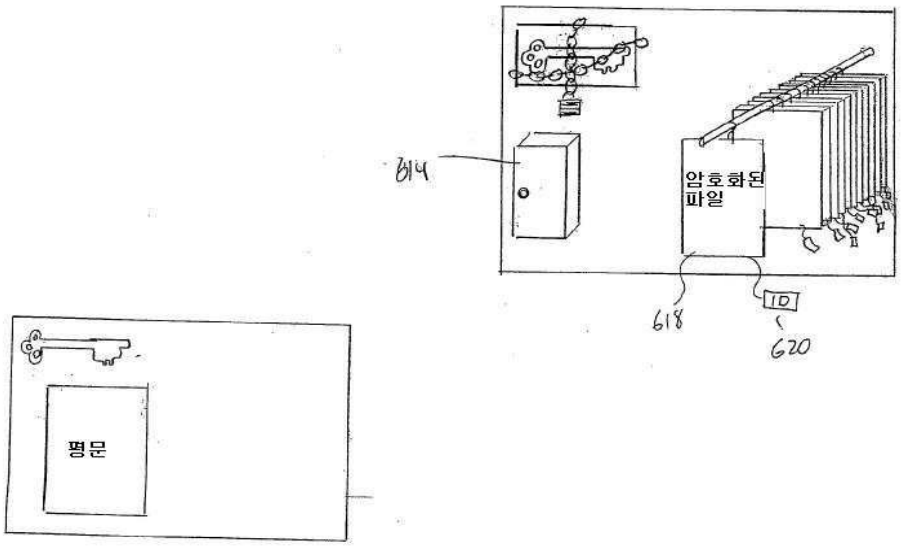
도면6d



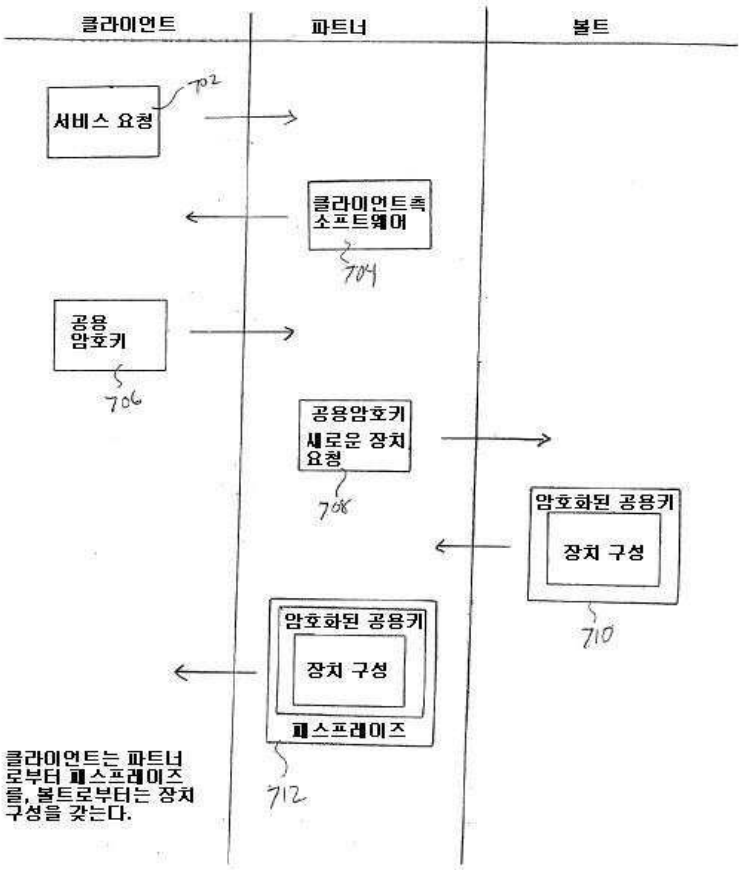
도면6e



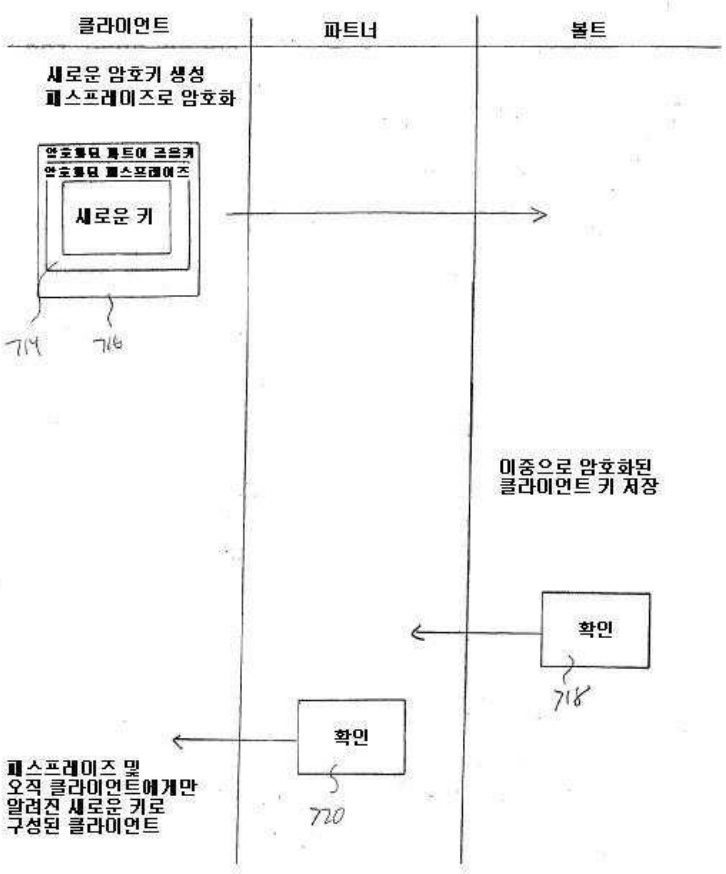
도면6f


614
암호화된
파일
618
ID
620
평문

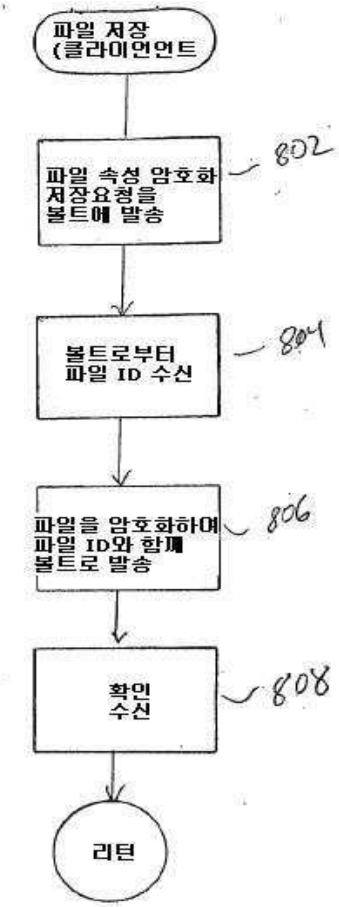
도면7a


클라이언트
파트너
볼트
서비스 요청
702
클라이언트측
소프트웨어
704
공용
암호키
706
공용암호키
새로운 장치
요청
708
암호화된 공용키
장치 구성
710
암호화된 공용키
장치 구성
패스프레이즈
712
클라이언트는 파트너
로부터 패스프레이즈
를, 볼트로부터는 장치
구성을 갖는다.

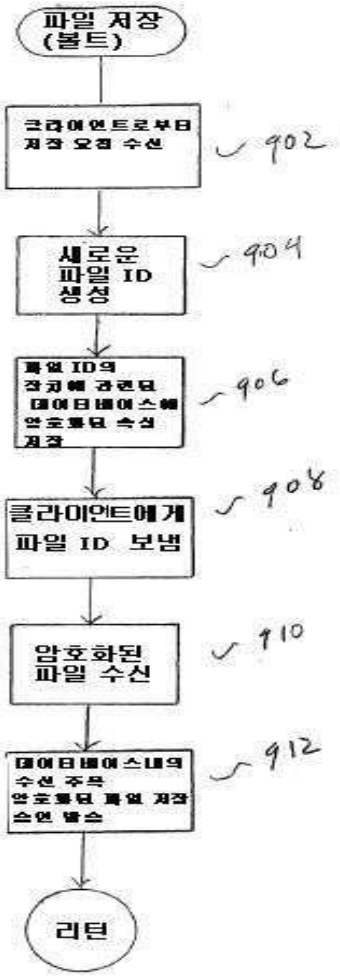
도면7b


클라이언트
파트너
볼트
새로운 암호키 생성
패스프레이즈로 암호화
암호화된 패트에 코드키
암호화된 패스프레이즈
새로운 키
714
716
이중으로 암호화된
클라이언트 키 저장
확인
718
확인
720
패스프레이즈 및
오직 클라이언트에게만
알려진 새로운 키로
구성된 클라이언트

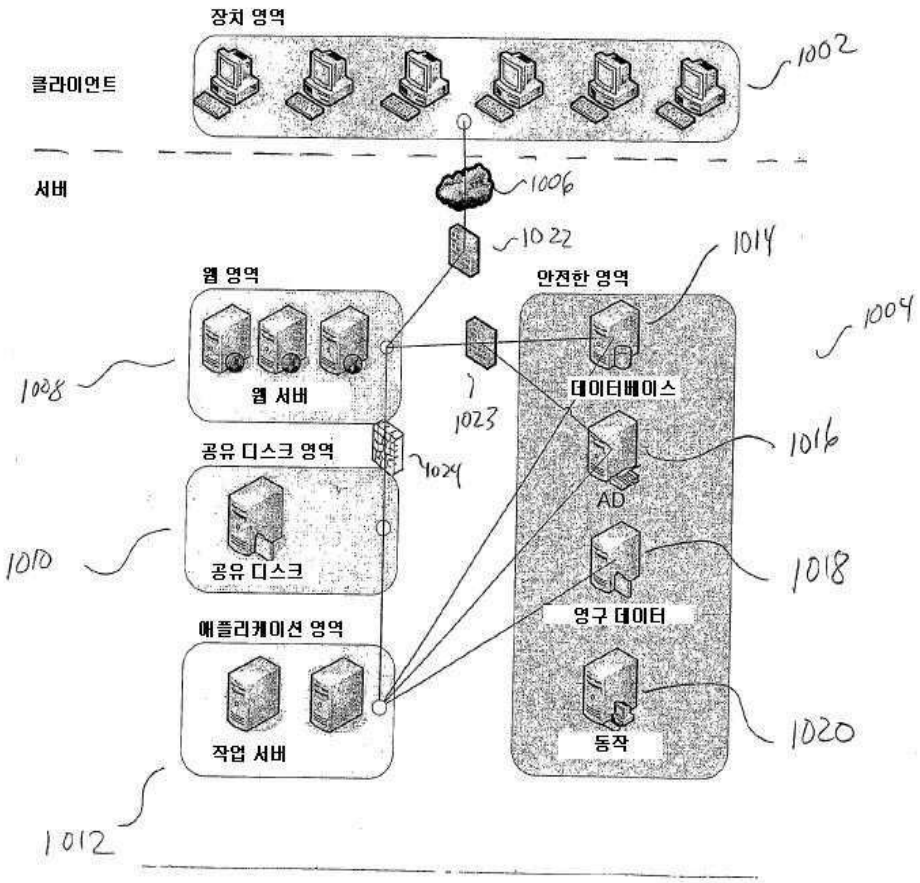
도면8


파일 저장
(클라이언트
파일 속성 암호화
저장요청을
볼트에 발송
802
볼트로부터
파일 ID 수신
804
파일을 암호화하여
파일 ID와 함께
볼트로 발송
806
확인
수신
808
리턴

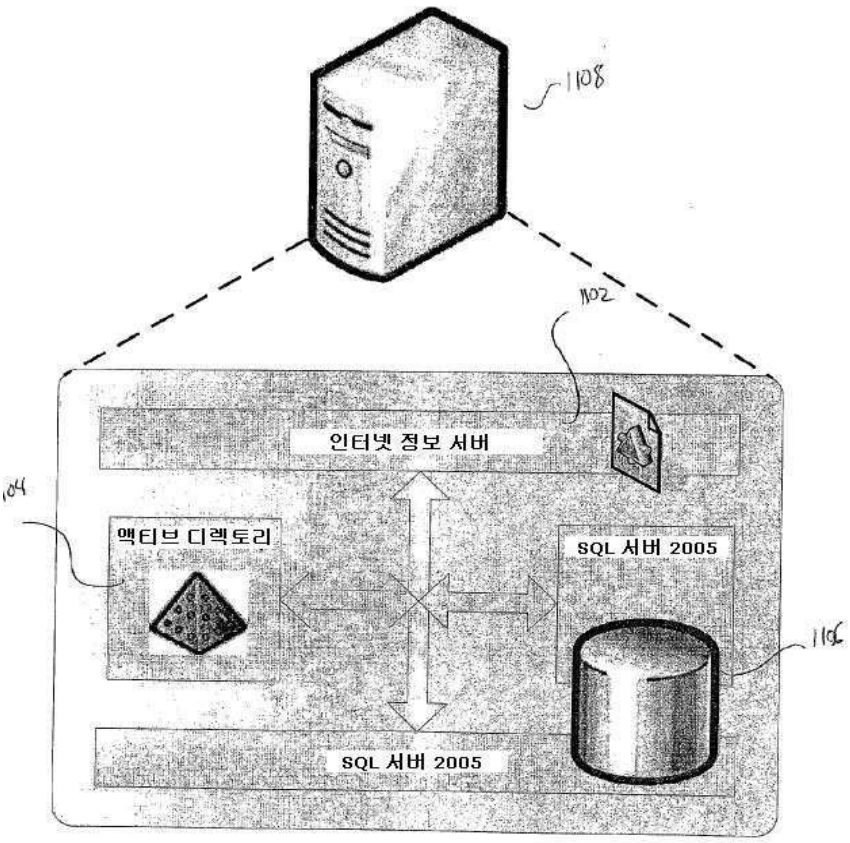
도면9


파일 저장 (볼트)
클라이언트로부터 저장 요청 수신
902
새로운 파일 ID 생성
904
파일 ID와 관련된 데이터베이스에 암호화된 속성 저장
906
클라이언트에게 파일 ID 보냄
908
암호화된 파일 수신
910
데이터베이스내의 수신 주소 암호화된 파일 저장 스텐 넘스
912
리턴

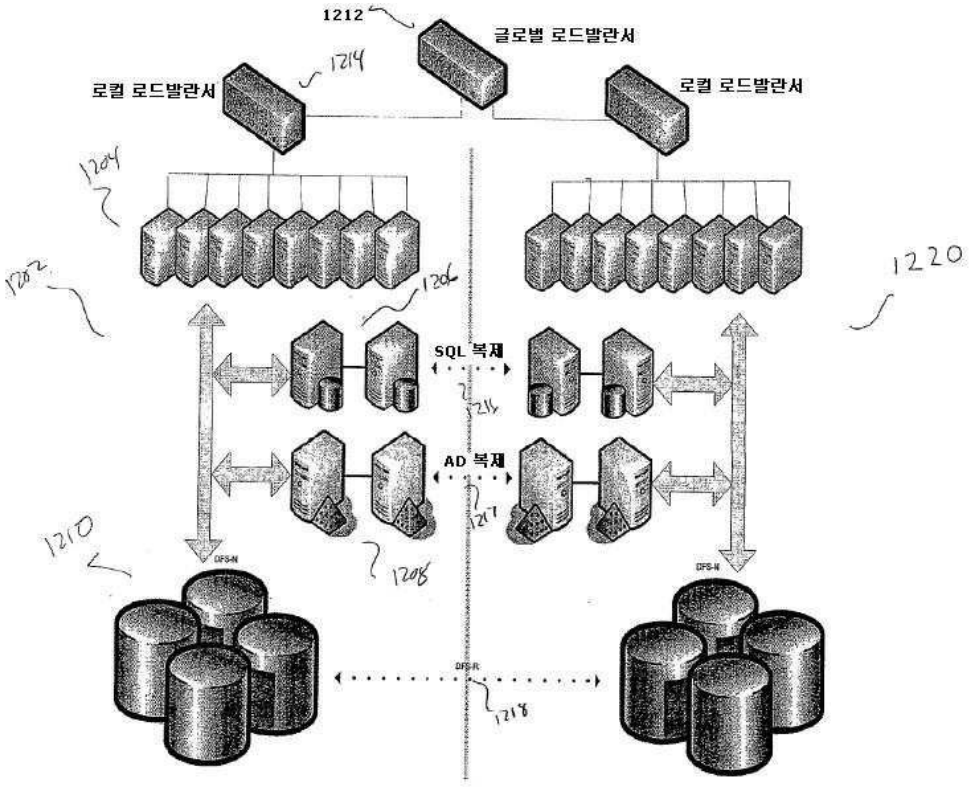
도면10


장치 영역
클라이언트
1002
서버
1006
1022
1014
웹 영역
안전한 영역
1004
1008
웹 서버
1023
데이터베이스
공유 디스크 영역
1024
1016
AD
1010
공유 디스크
1018
영구 데이터
애플리케이션 영역
1020
동작
작업 서버
1012

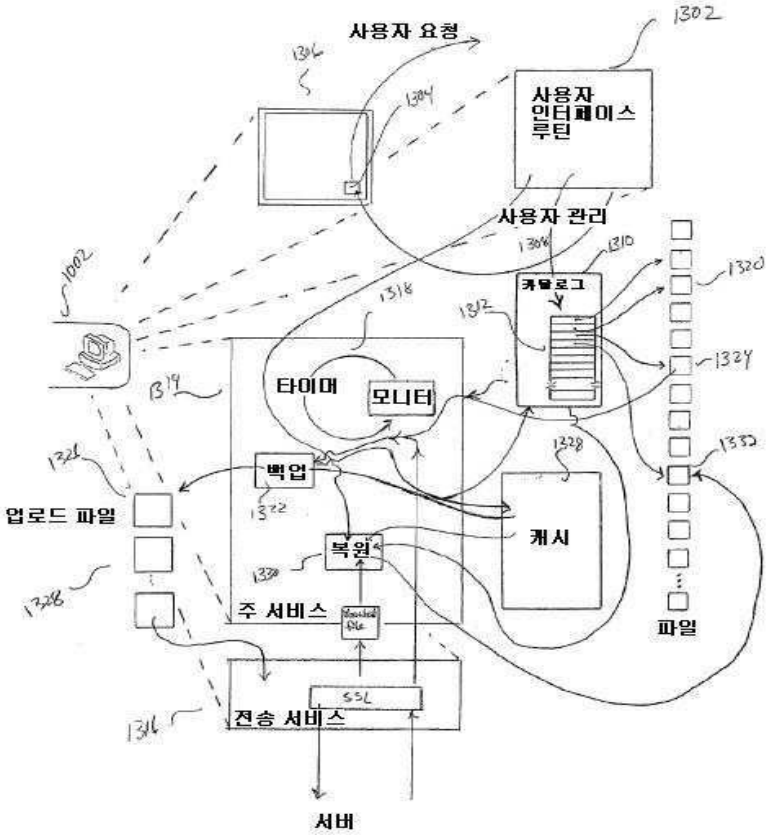
도면11


1108
1102
인터넷 정보 서버
104
액티브 디렉토리
SQL 서버 2005
1106
SQL 서버 2005

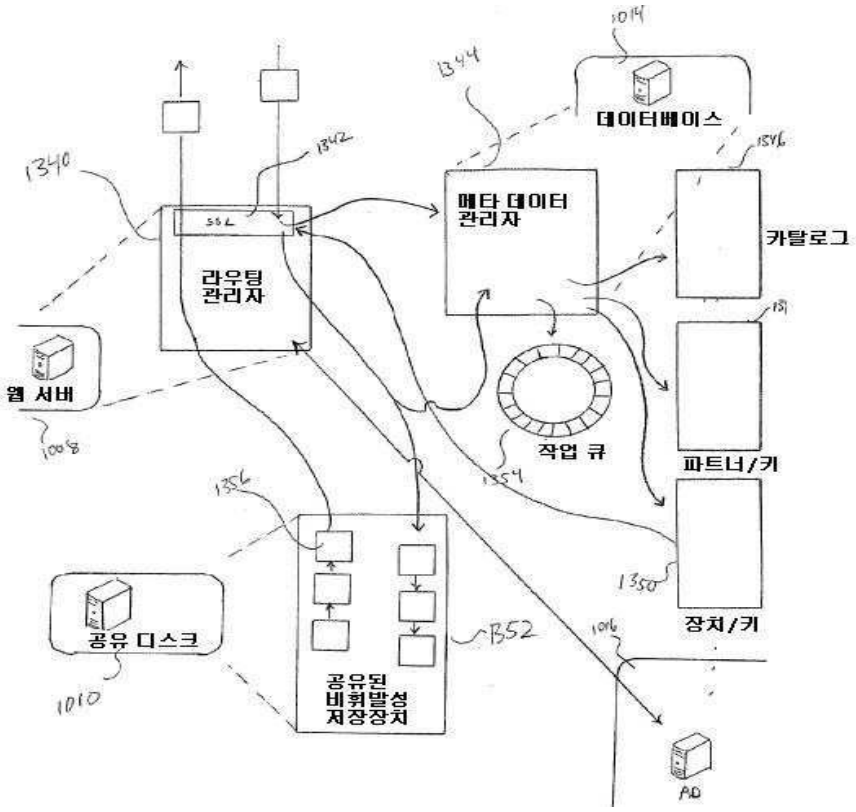
도면12


1212
글로벌 로드발란서
1214
로컬 로드발란서
로컬 로드발란서
1204
1202
1206
1220
SQL 복제
1216
AD 복제
1217
1210
1208
DFS-N
DFS-N
DFS-R
1218

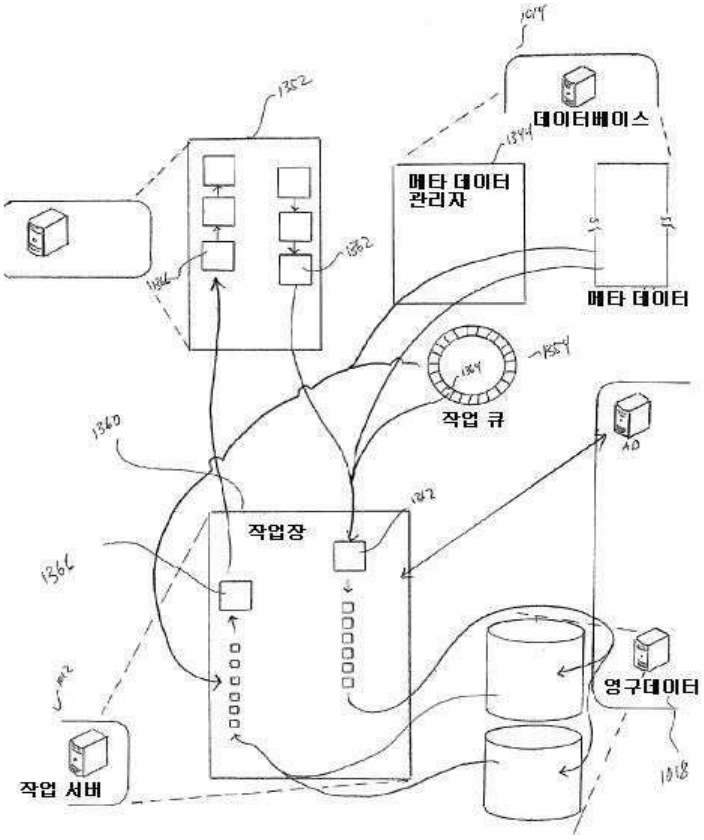
도면13a


사용자 요청
1306
1304
1302
사용자 인터페이스 루틴
사용자 관리
1308
1310
1320
1316
1312
1324
1002
1314
타이머
모니터
1328
1332
1321
벽업
1322
업로드 파일
캐시
복원
1330
1326
주 서비스
파일
SSL
전송 서비스
서버

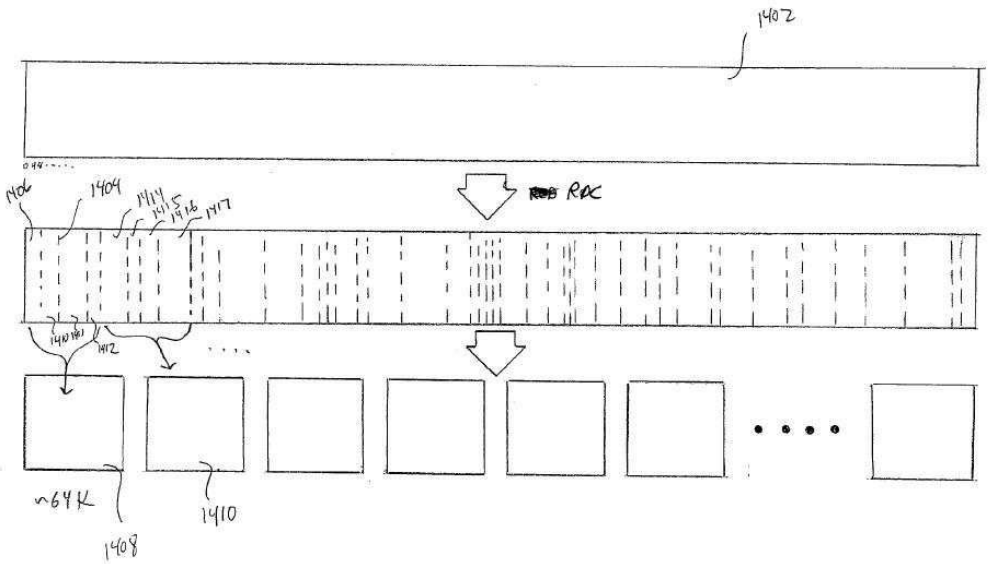
도면13b


데이터베이스
메타 데이터
관리자
카탈로그
라우팅
관리자
웹 서버
작업 큐
파트너/키
장치/키
공유 디스크
공유된
네워발성
저장장치
AD

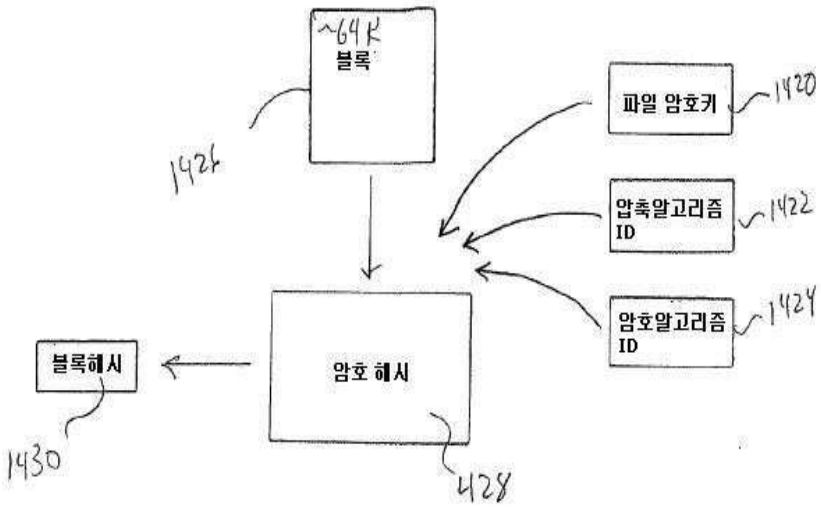
도면13c


데이터베이스
메타 데이터
관리자
메타 데이터
작업 큐
AD
작업장
영구데이터
작업 서버

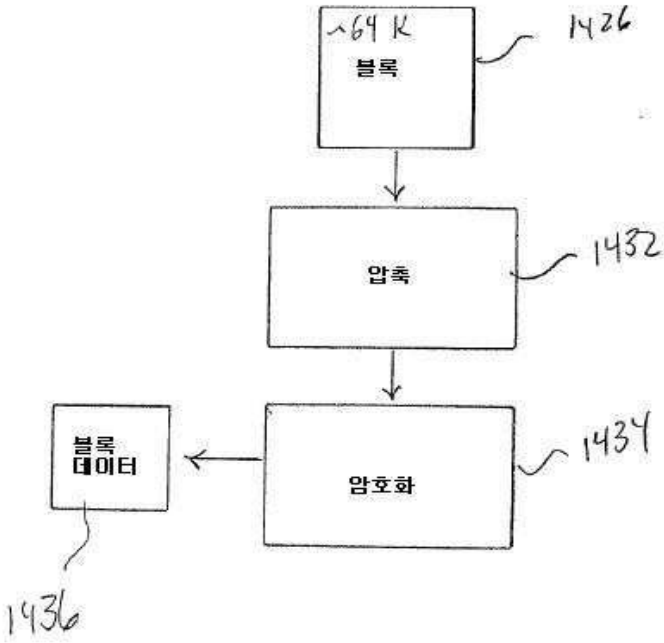
도면14a


1402
1404
1406
1414
1416
1417
RAC
1412
~64K
1408
1410

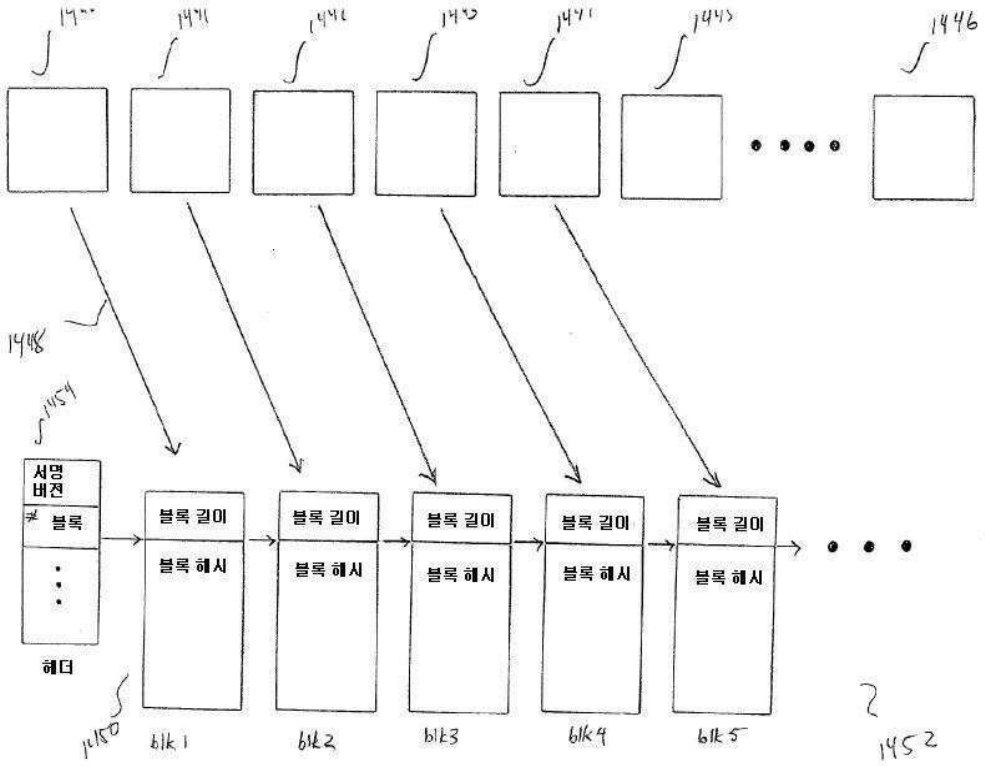
도면14b


~64K
블록
1426
파일 암호키
1420
압축알고리즘
ID
1422
암호알고리즘
ID
1424
암호 해시
블록해시
1430
428

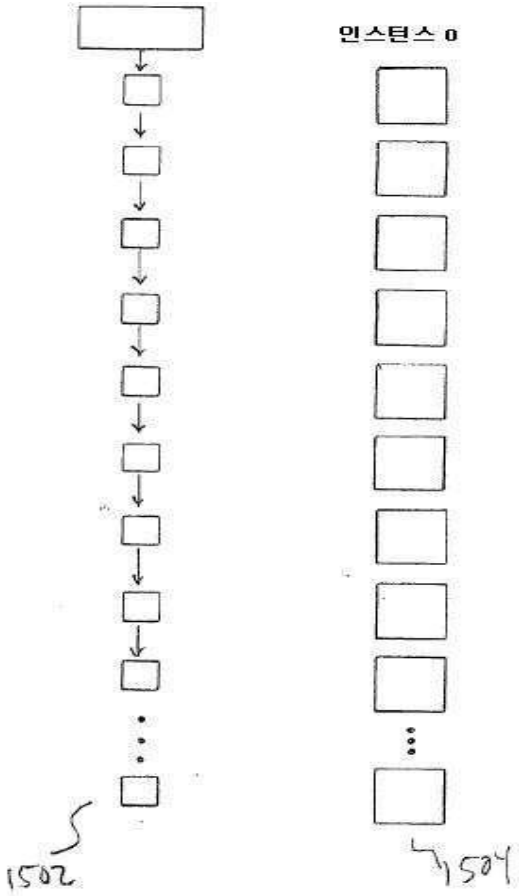
도면14c


~64 K
블록
1426
압축
1432
블록
데이터
암호화
1434
1436

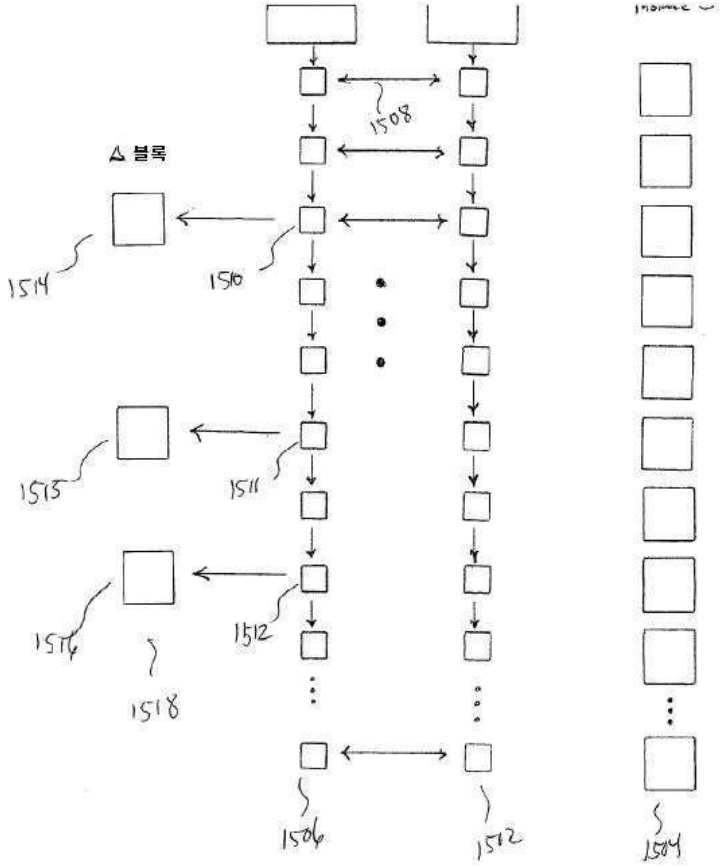
도면14d


1446
1448
1454
서명
버전
블록
헤더
블록 길이
블록 헤시
1450
blk 1
blk 2
blk 3
blk 4
blk 5
1452

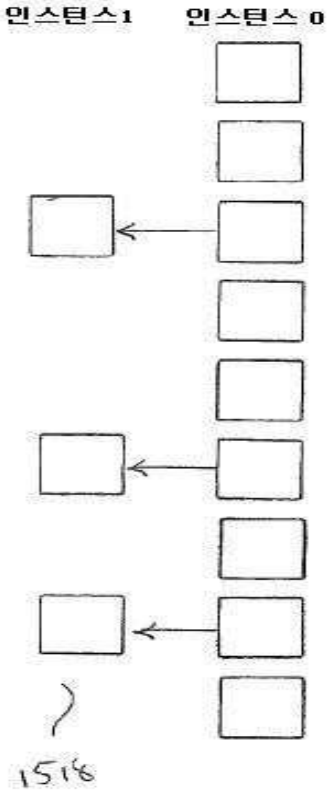
도면15a



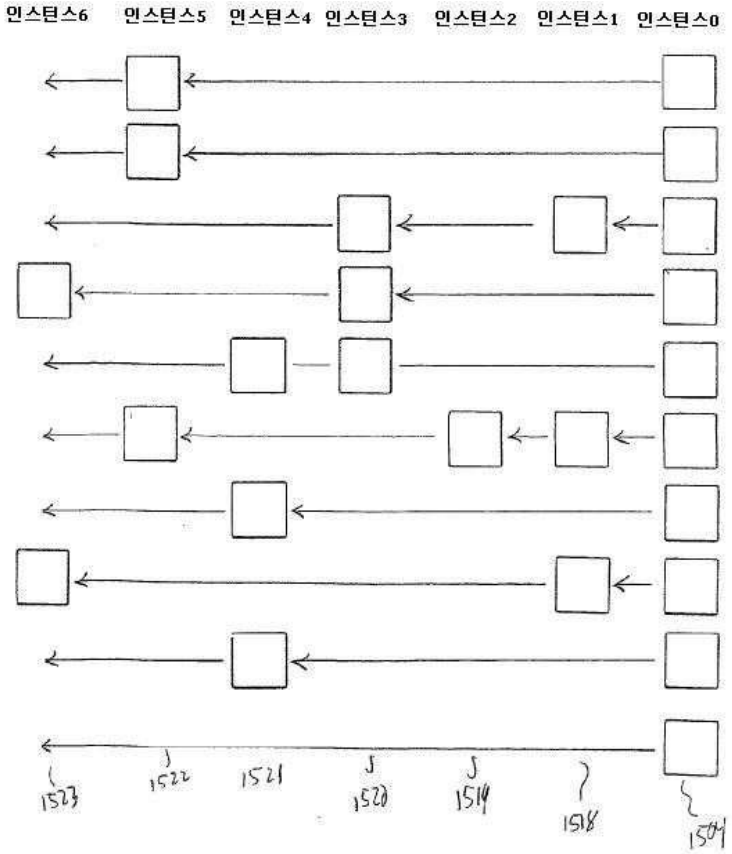
도면15b



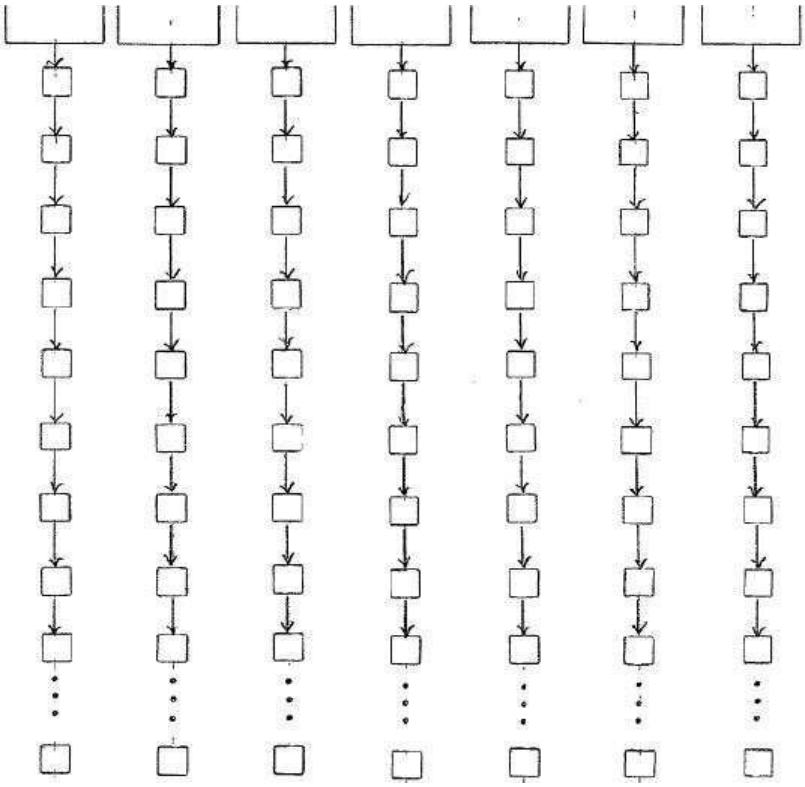
도면15c


인스턴스1
인스턴스 0
1518

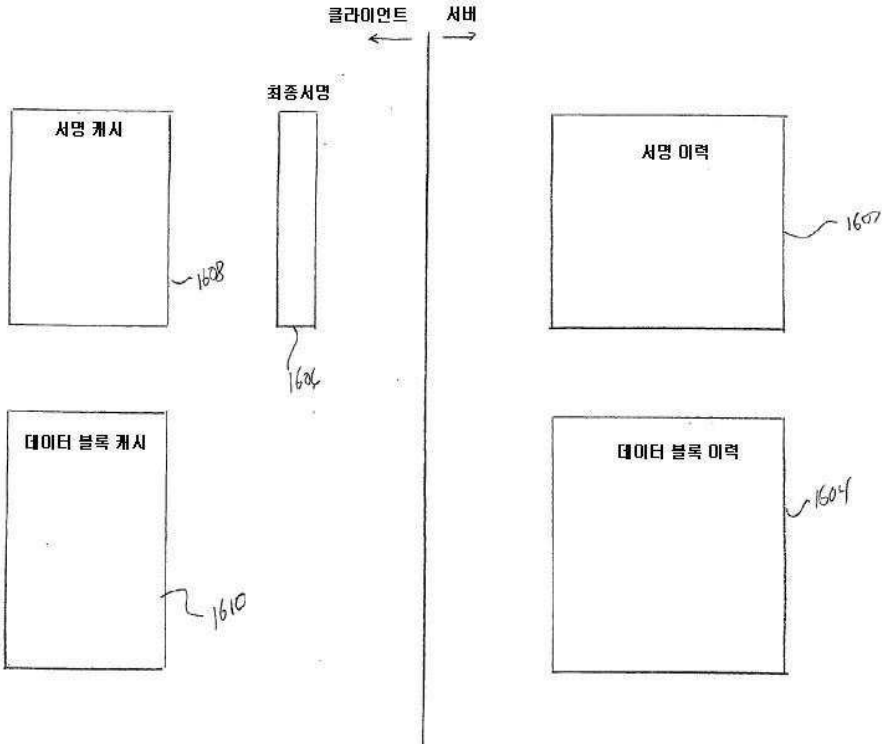
도면15d


인스턴스6
인스턴스5
인스턴스4
인스턴스3
인스턴스2
인스턴스1
인스턴스0
1523
1522
1521
1520
1514
1518
1504

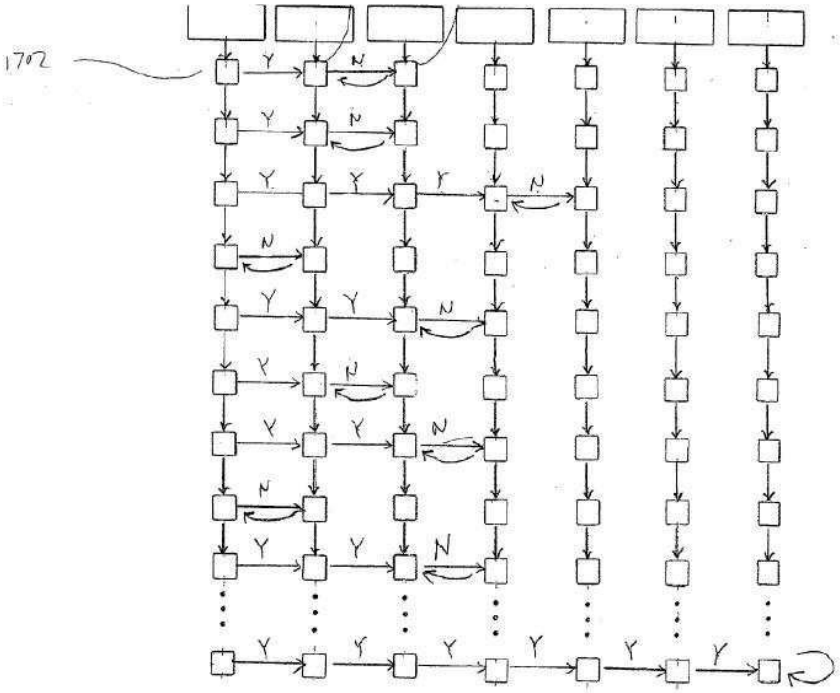
도면15e



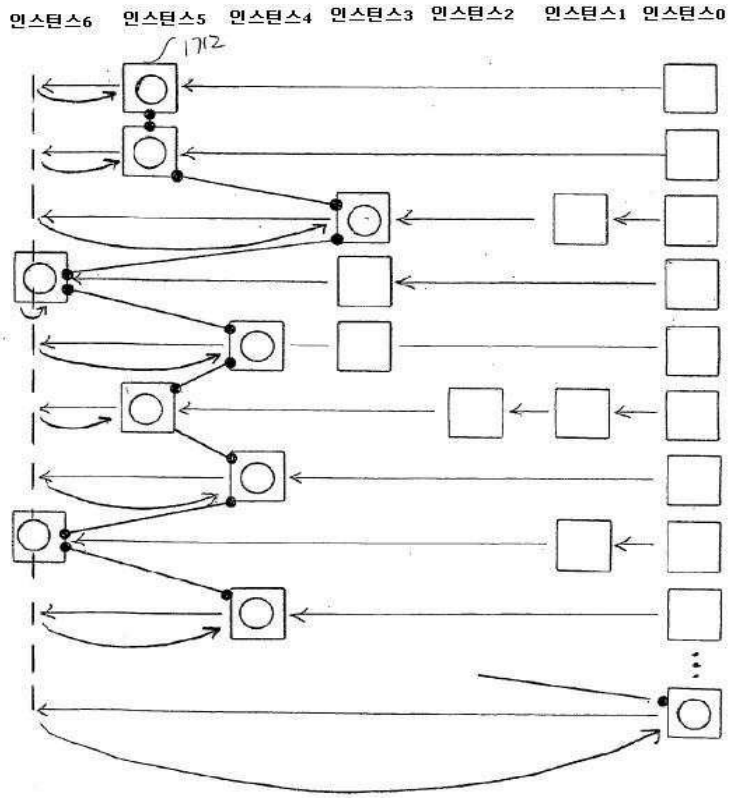
도면16


클라이언트
서버
최종서명
서명 캐시
1608
서명 이력
1607
1606
데이터 블록 캐시
1610
데이터 블록 이력
1604

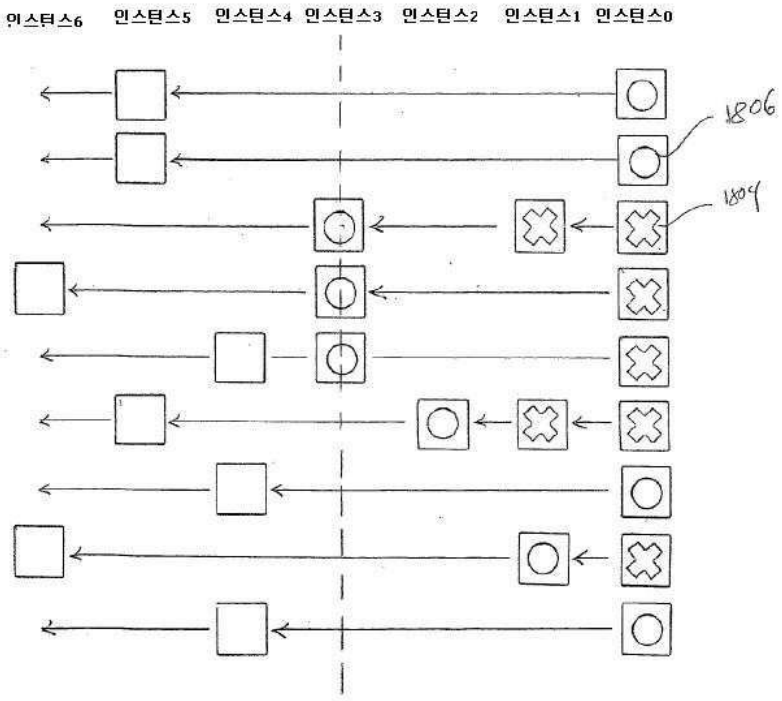
도면17a



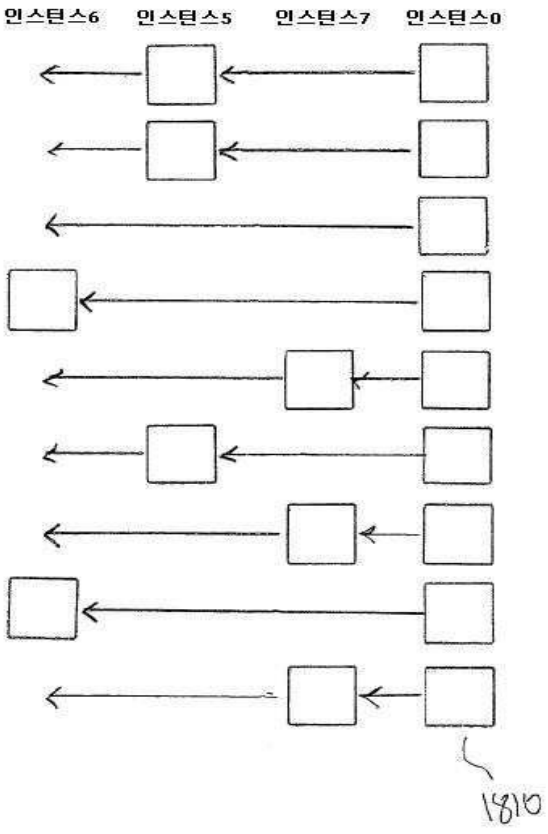
도면17b



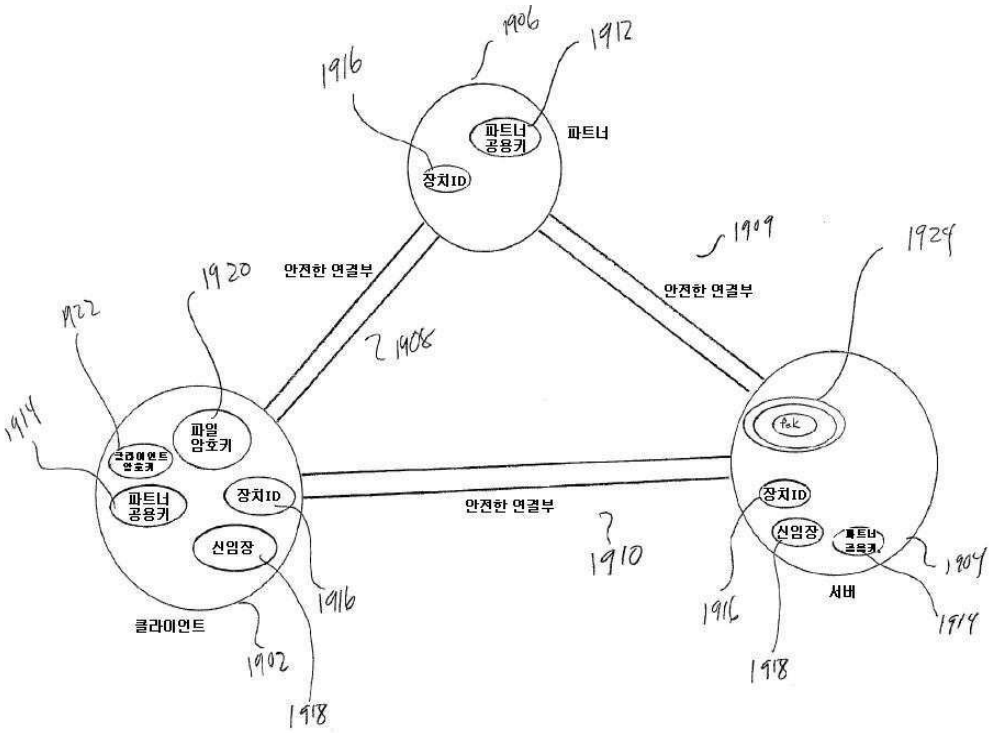
도면18a


인스턴스6
인스턴스5
인스턴스4
인스턴스3
인스턴스2
인스턴스1
인스턴스0
1806
1804

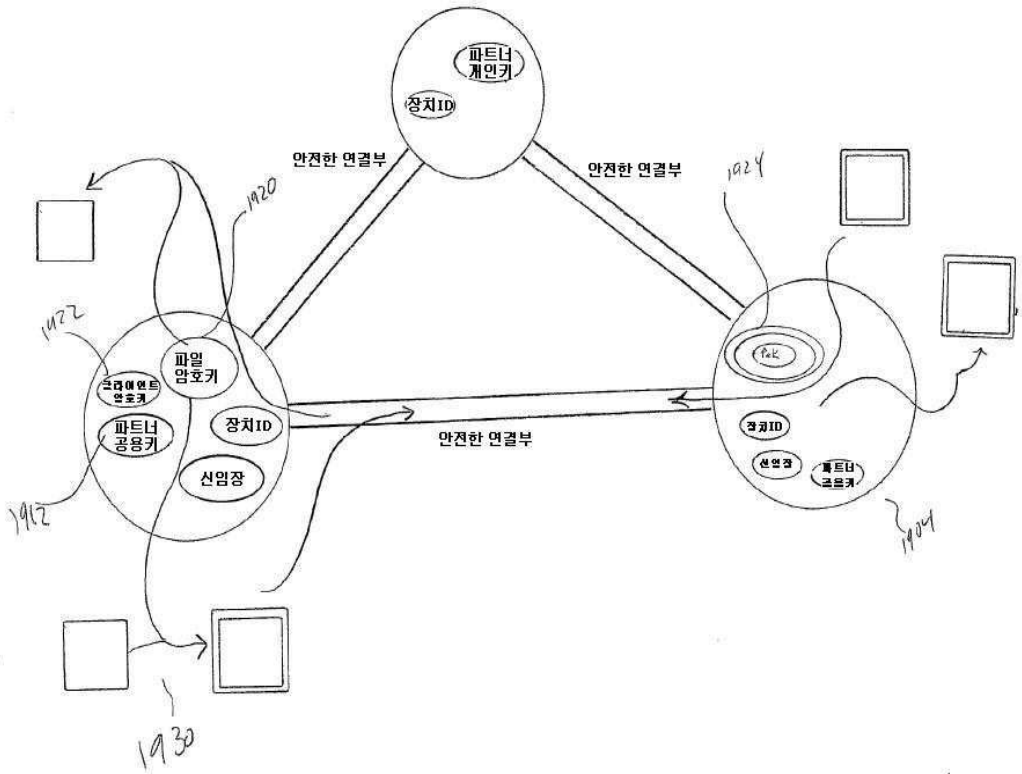
도면18b


인스턴스6
인스턴스5
인스턴스7
인스턴스0
1810

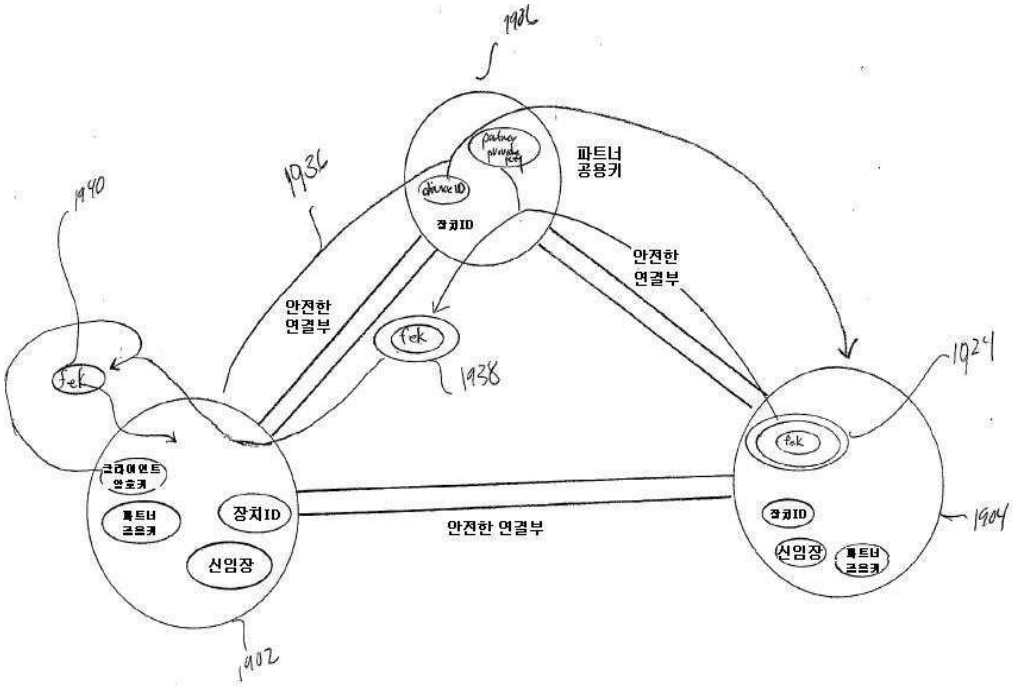
도면19a


1906
1912
1916
파트너 공용키
파트너
장치ID
1909
1924
1920
안전한 연결부
안전한 연결부
1922
1908
1914
파일 암호키
클라이언트 암호키
파트너 공용키
장치ID
안전한 연결부
1910
신임장
1916
서버
장치ID
신임장
파트너 공용키
1904
1914
클라이언트
1902
1918
1918

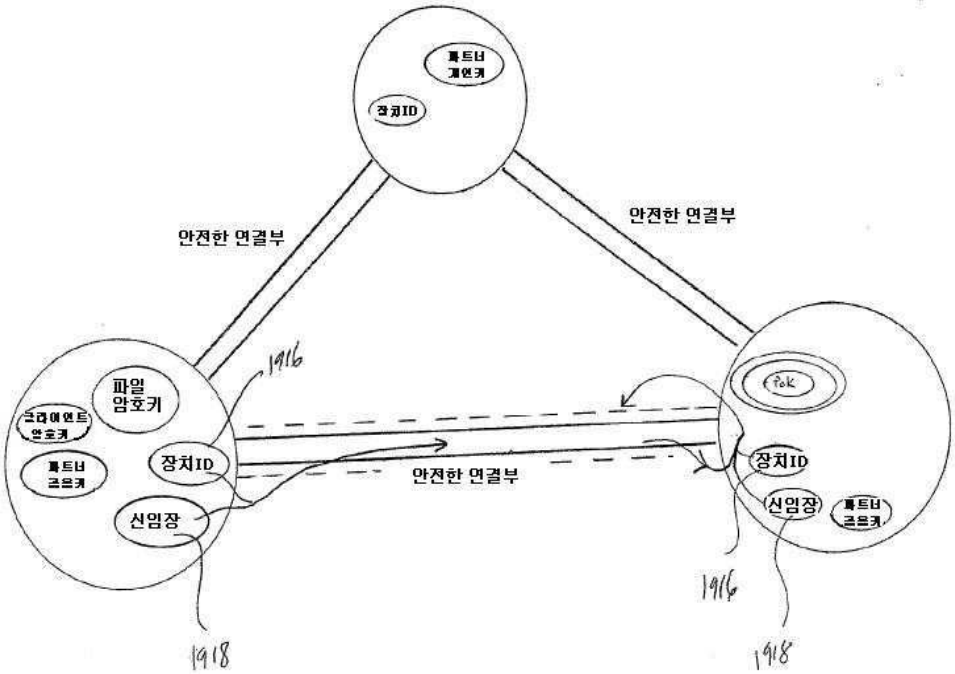
도면19b


파트너 개인키
장치ID
안전한 연결부
안전한 연결부
1924
1920
1922
파일 암호키
클라이언트 암호키
파트너 공용키
장치ID
신임장
안전한 연결부
장치ID
신임장
파트너 공용키
1912
1904
1930

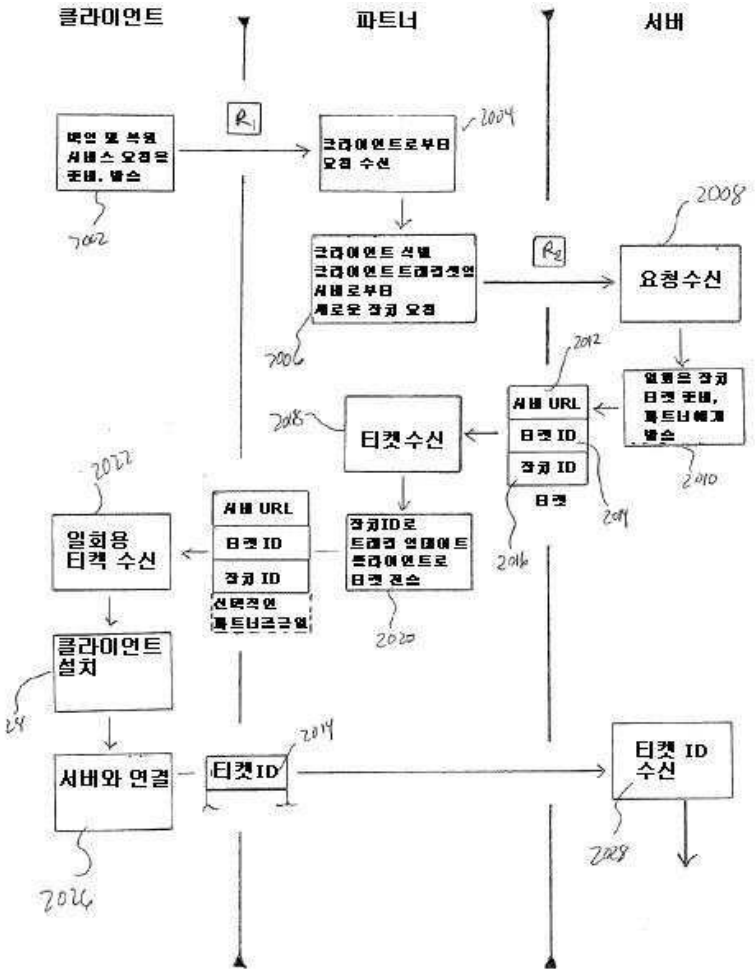
도면19c


1936
1940
파트너
공용키
안전한
연결부
안전한
연결부
1938
1924
1904
안전한 연결부
장치ID
신임장
1902

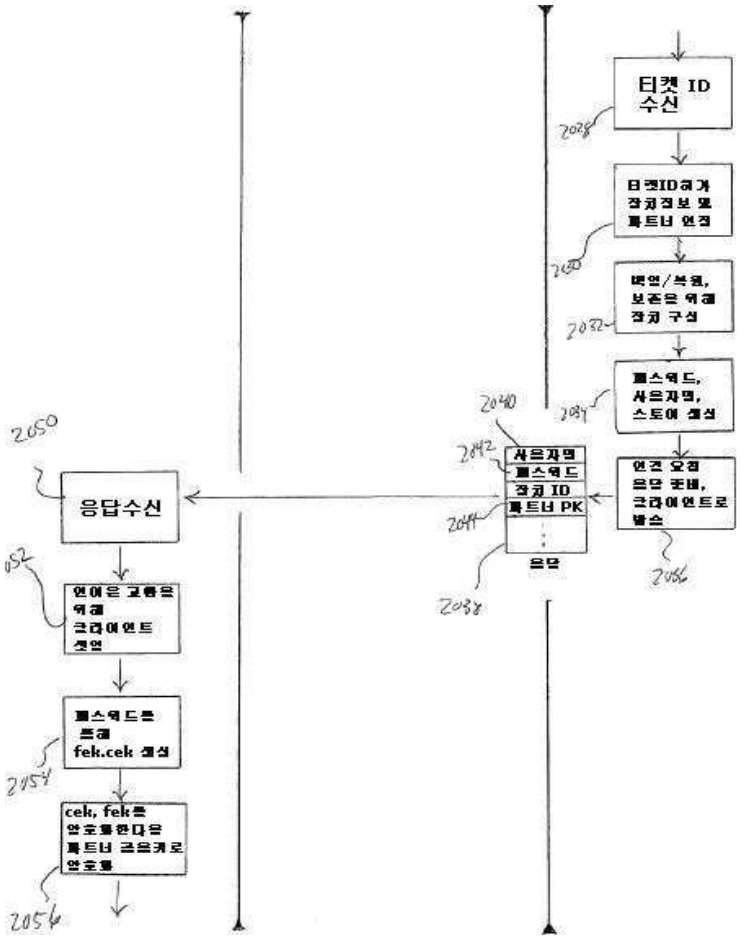
도면19d


장치ID
안전한 연결부
안전한 연결부
1916
파일
암호키
장치ID
신임장
안전한 연결부
장치ID
신임장
1916
1918
1918

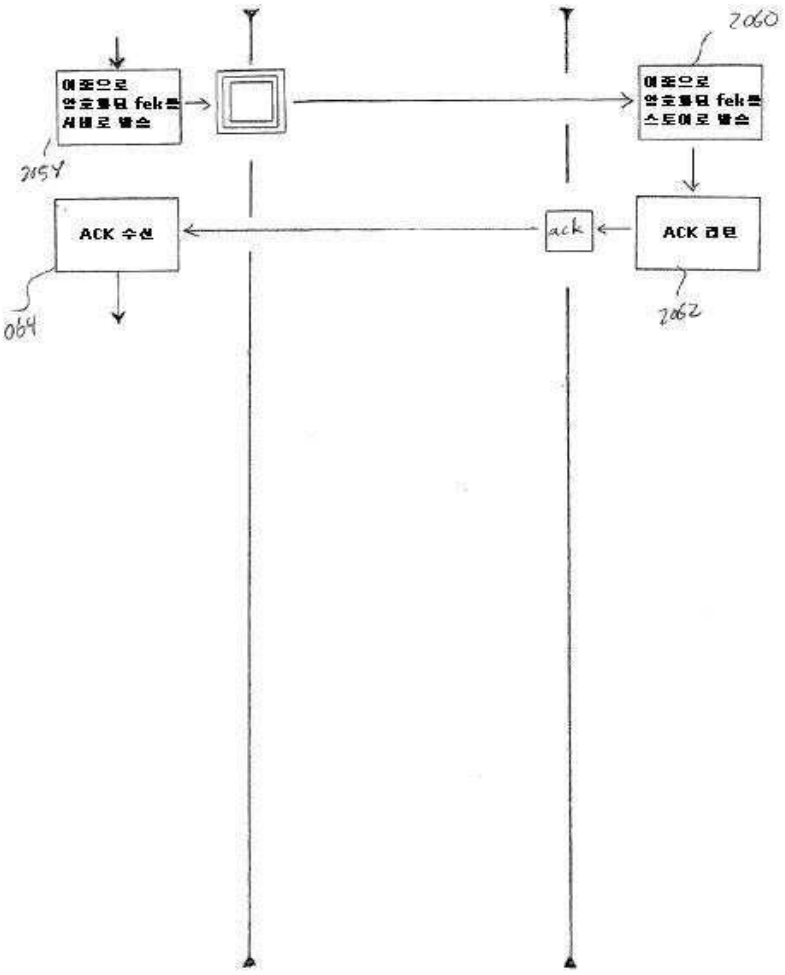
도면20a


클라이언트
파트너
서버
R1
R2
2004
2008
2012
2018
2022
2016
2020
2024
2014
2028
2026
2010
2009
요청수신
티켓수신
일회용 티켓 수신
클라이언트 설치
서버와 연결
티켓ID
티켓 ID 수신
서버 URL
티켓 ID
장치 ID
티켓

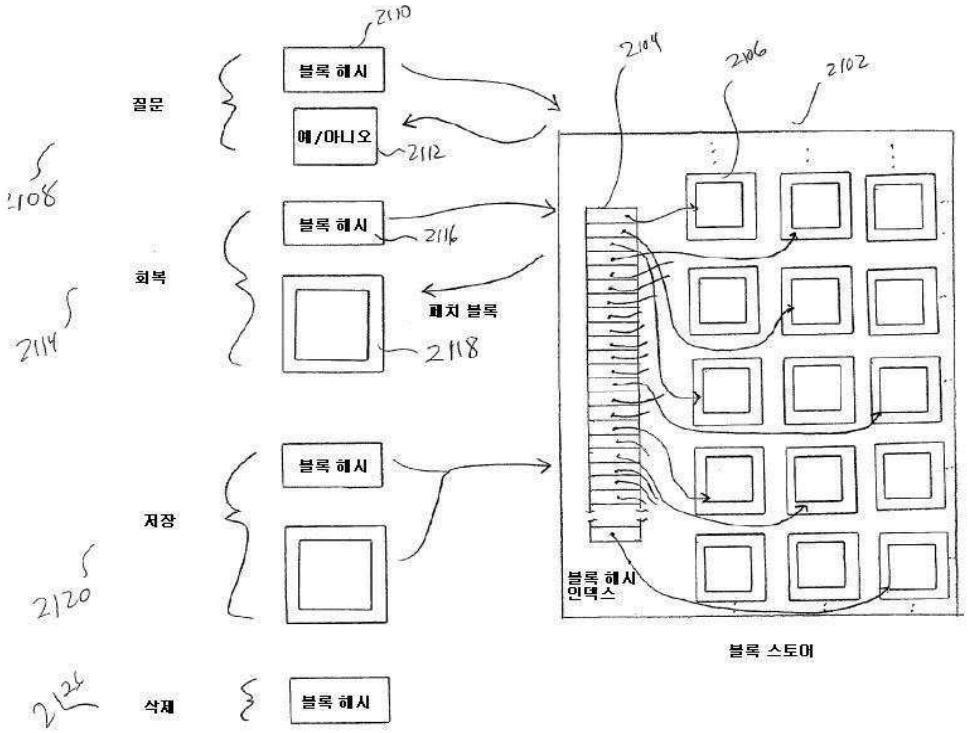
도면20b


티켓 ID 수신
2028
티켓ID에 해당 장치정보 및 파트너 인증
2030
백업/복원, 보관을 위해 장치 구성
2032
패스워드, 사용자명, 스토어 생성
2034
인증 오류 응답 준비, 클라이언트로 넘스
2036
2040
2042
사용자명
패스워드
장치 ID
파트너 PK
2044
응답
2038
2050
응답수신
052
인증을 고유한 위해 클라이언트 생성
패스워드를 통해 fek.cek 생성
2054
cek, fek를 암호화한다음 파트너 공유키로 암호화
2056

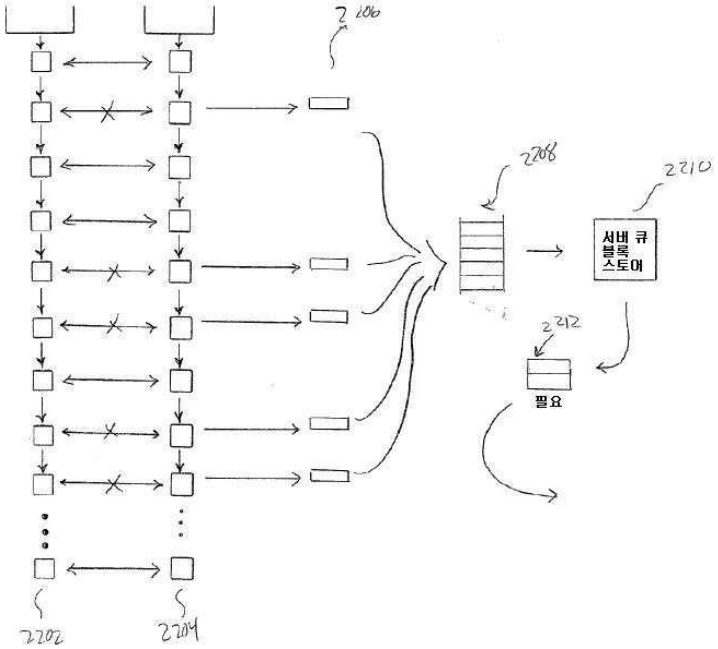
도면20c



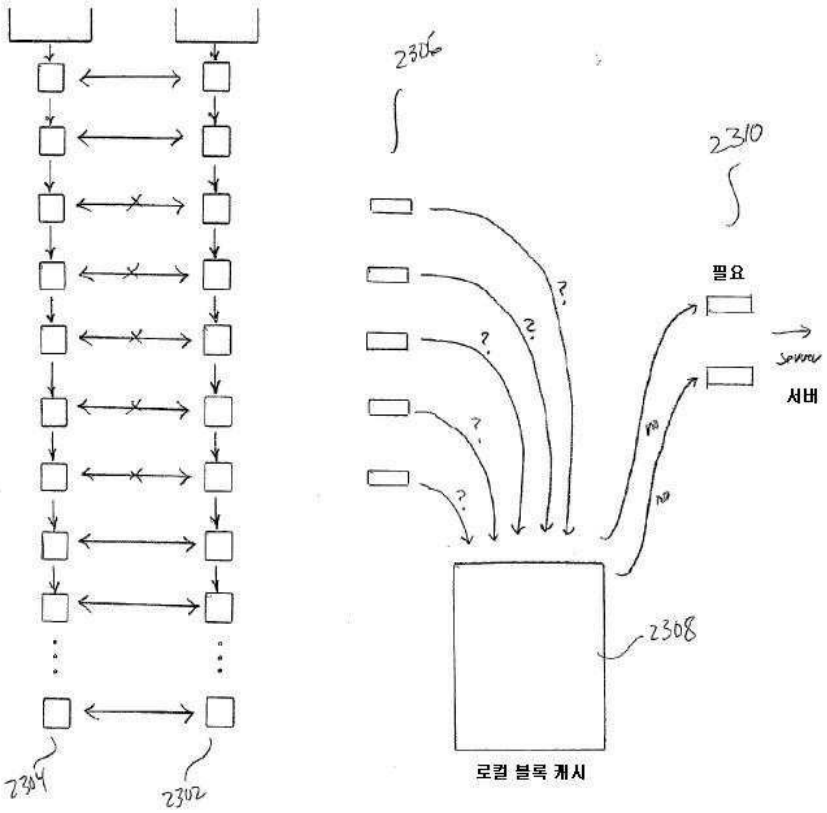
도면21


2110
블록 해시
질문
예/아니오
2112
2108
블록 해시
2116
회복
패치 블록
2114
2118
블록 해시
저장
2120
2126
삭제
블록 해시
2104
2106
2102
블록 해시
인덱스
블록 스토어

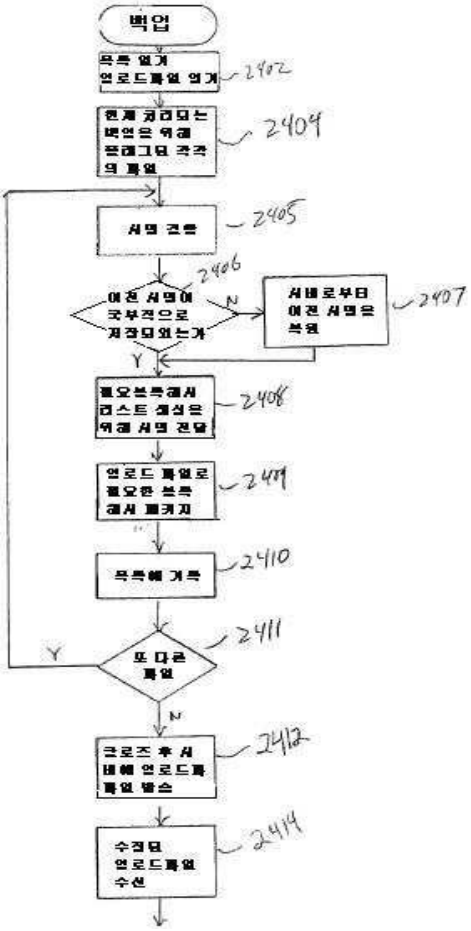
도면22


2106
2208
2210
서버 큐
블록
스토어
2212
필요
2202
2204

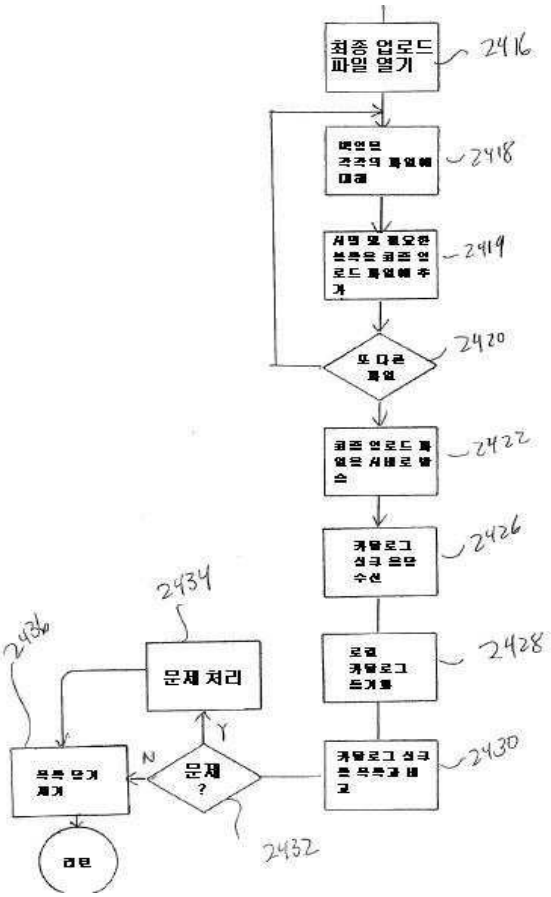
도면23



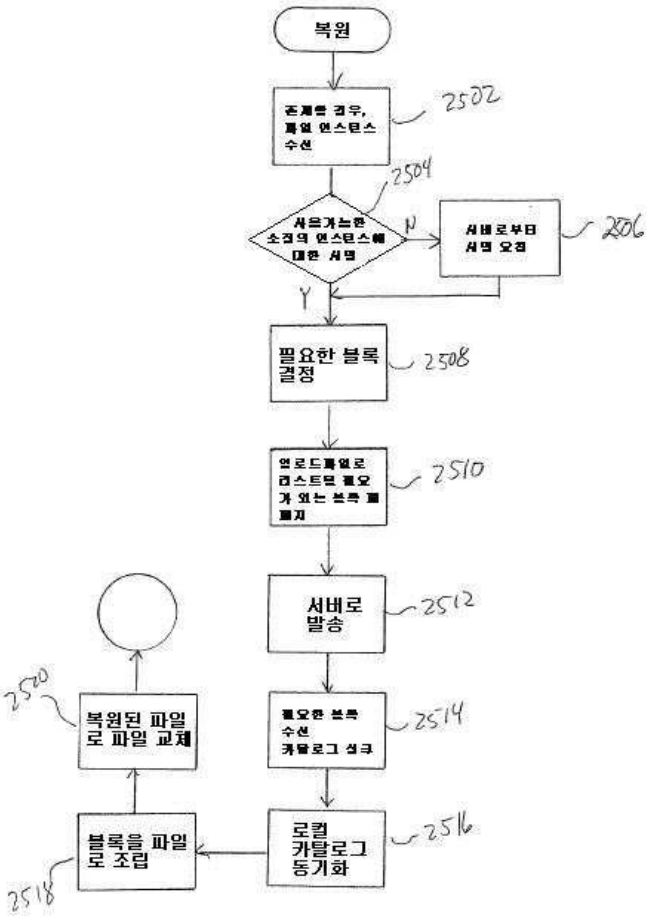
도면24a


백업
목록 얻기
업로드될 업기
2402
현재 처리되는
백업은 워크
플래그된 객체
의 목록
2404
서명 검증
2405
2406
이전 서명이
국부적으로
저장되었는가
N
서버로부터
이전 서명을
복원
2407
Y
필요블록레시
리스트 생성을
위해 서명 전달
2408
업로드 목록로
필요한 블록
레시 패키지
2409
목록에 기록
2410
또 다른
목록
2411
Y
N
크로즈 후 서
버에 업로드목
록을 발송
2412
수정된
업로드목록
수신
2414

도면24b


최종 업로드 파일 열기
2416
2418
2419
2420
또 다른 파일
2422
2426
2428
2430
2432
문제 ?
Y
N
문제 처리
2434
2436
리턴

도면25


복원
2502
2504
N
2506
Y
필요한 블록 결정
2508
2510
서버로 발송
2512
2514
로컬 카탈로그 등기화
2516
블록을 파일로 조립
2518
복원된 파일로 파일 교체
2520