(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년12월27일
(11) 등록번호 10-2618665
(24) 등록일자 2023년12월22일

(51) 국제특허분류(Int. Cl.)
G06F 21/64 (2013.01) *G06F 8/658* (2018.01)
G06F 8/71 (2018.01) *H04L 9/06* (2006.01)
(52) CPC특허분류
G06F 21/64 (2013.01)
G06F 8/658 (2018.02)
(21) 출원번호 10-2021-7036946
(22) 출원일자(국제) 2020년04월17일
심사청구일자 2021년11월11일
(85) 번역문제출일자 2021년11월11일
(65) 공개번호 10-2021-0151926
(43) 공개일자 2021년12월14일
(86) 국제출원번호 PCT/US2020/028788
(87) 국제공개번호 WO 2020/214968
국제공개일자 2020년10월22일
(30) 우선권주장
16/389,840 2019년04월19일 미국(US)
(56) 선행기술조사문헌
US10057243 B1*
US20180343175 A1*
US20190012595 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
사르코스 코퍼레이션
미국 84101 유타 솔트레이크시티 수트 150 웨스트 500 사우스 650
(72) 발명자
스미스, 프레이저 엠.
미국 유타주 84108 솔트 레이크 시티 와카라 웨이 360
(74) 대리인
특허법인 무한

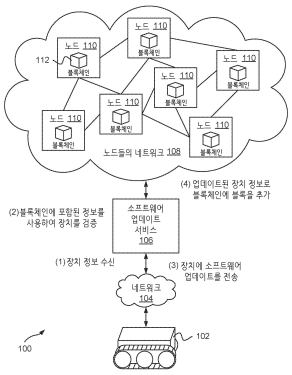
전체 청구항 수 : 총 17 항 심사관 : 구대성

(54) 발명의 명칭 **블록체인을 사용한 버전 이력 관리**

(57) 요 약

블록체인을 사용하여 디바이스의 소프트웨어 버전 이력을 관리하고 보호하는 기술에 대해 설명한다. 예시적인 방법은 소프트웨어 업데이트들을 수신하도록 구성된 장치에 대한 장치 정보를 수신하는 것을 포함할 수 있고, 장치에 대한 장치 이력은 다수의 네트워크 노드들 사이에 분산된 블록체인에 포함될 수 있다. 장치 정보는 장치의 아이덴터티를 확인하기 위해 블록체인에 포함된 장치 이력에 대해 검증될 수 있고, 장치에 대한 소프트웨어 업데이트가 개시될 수 있다. 장치에 소프트웨어 업데이트가 설치되었다는 표시를 수신하고 장치에 설치된 소프트웨어 업데이트에 대한 정보를 포함하는 블록을 생성할 수 있다. 블록은 블록의 수락을 결정하고 블록이 수락되는 경우 블록을 블록체인에 추가하도록 구성된 네트워크 노드로 전송할 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06F 8/71 (2013.01)
H04L 9/0637 (2013.01)
H04L 9/50 (2022.05)

명 세 서

청구범위

청구항 1

컴퓨터로 구현되는 방법에 있어서,

소프트웨어 업데이트의 출시에 응답하여, 블록체인으로부터 복수의 장치들에 대한 장치 이력들을 획득하는 단계

상기 복수의 장치들 중 상기 소프트웨어 업데이트를 수신할 자격이 있는 장치를 식별하기 위하여, 상기 블록체인으로부터 획득된 상기 장치 이력들을 참조하는 단계 - 상기 블록체인은 복수의 네트워크 노들들에 분산되어 있고, 상기 복수의 장치들은 소프트웨어 업데이트들을 수신하도록 구성됨 -;

상기 복수의 장치들 중 상기 소프트웨어 업데이트를 수신할 자격이 있는 것으로 식별된 장치에 대한 장치 정보를 획득하는 단계;

상기 장치의 아이덴터티를 확인하기 위해, 상기 블록체인으로부터 획득된 상기 장치에 대한 장치 이력에 대하여, 상기 장치 정보를 검증하는 단계;

상기 장치에 상기 소프트웨어 업데이트들을 설치하도록 구성된 상기 장치에 대한 소프트웨어 업데이트를 개시하는 단계;

상기 장치에 상기 소프트웨어 업데이트가 설치되었다는 표시를 수신하는 단계; 및

상기 장치에 설치된 상기 소프트웨어 업데이트에 대한 정보 및 장치 정보를 포함하는 블록의 생성을 개시하는 단계

를 포함하고,

상기 블록의 수락을 결정하고, 상기 블록이 수락된 경우 상기 블록을 상기 블록체인에 추가하기 위해, 상기 블록은, 상기 복수의 네트워크 노드들로 전송되는,

방법.

청구항 2

제1항에 있어서,

상기 장치 이력을 획득하는 단계는,

상기 블록체인으로부터 상기 장치에 대한 상기 장치 이력을 획득하는 단계;

상기 장치의 상기 아이덴터티를 확인하기 위하여, 상기 장치 이력에 기록된 장치 식별자에 대하여 상기 장치 정보에 포함된 장치 식별자를 검증하는 단계; 및

상기 장치에 현재 설치된 소프트웨어의 버전을 확인하기 위하여, 상기 장치 이력에 기록된 소프트웨어 버전에 대하여 상기 장치 정보에 포함된 소프트웨어 버전 식별자를 체크하는 단계

를 더 포함하는,

방법.

청구항 3

제1항에 있어서,

상기 블록체인으로부터 상기 소프트웨어 업데이트에 대한 소프트웨어 서명을 획득하는 단계; 및

상기 장치에 상기 소프트웨어 서명을 제공하는 단계

를 더 포함하고,

상기 장치는,

상기 소프트웨어 서명을 사용하여 상기 소프트웨어 업데이트의 사본을 검증하도록 구성된,

방법.

청구항 4

제1항에 있어서,

상기 소프트웨어 업데이트를 개시하는 단계는,

상기 소프트웨어 업데이트를 상기 장치에 설치하라는 명령어들과 함께 상기 소프트웨어 업데이트와 연관된 소프트웨어 사본을 상기 장치에 전송하는 단계

를 더 포함하는,

방법.

청구항 5

제1항에 있어서,

상기 소프트웨어 업데이트를 개시하는 단계는,

상기 네트워크 위치로부터 소프트웨어 사본을 획득하고 상기 소프트웨어 사본을 상기 장치에 설치하라는 명령어들과 함께 상기 소프트웨어 사본의 네트워크 위치를 상기 장치에 전송하는 단계

를 더 포함하는,

방법.

청구항 6

제1항에 있어서,

상기 소프트웨어 업데이트를 개시하는 단계는,

상기 소프트웨어 업데이트가 상기 장치에 설치되었음을 나타내는 업데이트된 장치 정보를 상기 복수의 네트워크 노드들에 전송하는 단계; 및

어떤 네트워크 노드가 상기 장치에 설치된 상기 소프트웨어 업데이트에 대한 상기 정보 및 상기 장치 정보를 포함하는 블록을 생성하는지 결정하기 위하여, 상기 업데이트된 장치 정보를 수신한 것에 응답하여, 상기 복수의 네트워크 노드들에 의하여, 작업 증명 처리를 수행하는 단계

를 더 포함하는,

방법.

청구항 7

제1항에 있어서,

상기 소프트웨어 업데이트를 개시하는 단계는,

상기 블록체인에 가장 최근에 추가된 블록에 대한 이전 블록 식별자를 상기 블록체인으로부터 획득하는 단계; 및

상기 블록에 상기 이전 블록 식별자를 저장하는 단계

를 더 포함하고,

블록 식별자는,

블록에 포함된 정보에 해시 함수를 적용하는 것에 의해 생성되는,

방법.

청구항 8

제7항에 있어서,

상기 블록에 포함된 정보에 상기 해시 함수를 적용하는 것에 의해, 상기 블록에 대한 현재 블록 식별자(current block identifier)를 생성하는 단계; 및

상기 블록에 상기 현재 블록 식별자를 저장하는 단계

를 더 포함하는,

방법.

청구항 9

제1항에 있어서,

상기 복수의 네트워크 노드들은,

피어 노드들의 공중 네트워크 또는 피어 노드들의 사설 네트워크 중 하나에 포함되는,

방법.

청구항 10

제1항에 있어서,

상기 장치는,

소프트웨어 업데이트들을 수신하도록 구성된 로봇 장치인,

방법.

청구항 11

제1항에 있어서,

상기 장치 이력들과 연관된 상기 복수의 장치들 중 상기 소프트웨어 업데이트를 수신할 자격이 있는 장치를 식별하는 것은,

상기 소프트웨어 업데이트를 수신할 장치의 자격(eligibility)을 결정하는 복수의 요소들과 상기 장치 이력들을 비교하는 것

을 더 포함하는, 방법.

청구항 12

시스템에 있어서,

적어도 하나의 프로세서, 및

명령어들을 포함하는 메모리 장치

를 포함하고,

상기 명령어들은, 상기 적어도 하나의 프로세서에 의해 실행되는 경우, 상기 시스템으로 하여금,

소프트웨어 업데이트의 출시에 응답하여, 블록체인으로부터 복수의 장치들에 대한 장치 이력들을 획득하도록 하고,

상기 복수의 장치들 중 상기 소프트웨어 업데이트를 수신할 자격이 있는 장치를 식별하기 위하여, 상기 블록체인으로부터 획득된 상기 장치 이력들을 참조하도록 하고 - 상기 블록체인은 복수의 네트워크 노들들에 분산되어 있고, 상기 복수의 장치들은 소프트웨어 업데이트들을 수신하도록 구성됨 -,

상기 복수의 장치들 중 상기 소프트웨어 업데이트를 수신할 자격이 있는 것으로 식별된 장치에 대한 장치 정보를 획득하도록 하고 - 상기 장치 정보는, 장치 식별자 및 상기 장치에 현재 설치된 소프트웨어의 버전을 포함함

-,

상기 장치의 아이덴터티 및 상기 장치에 현재 설치된 소프트웨어의 버전을 확인하기 위해, 상기 장치에 대한 장치 이력을 사용하여 상기 장치 정보를 검증하도록 하고,

상기 소프트웨어의 업데이트된 버전을 상기 장치에 전송하도록 하고 - 상기 장치는, 상기 장치에 상기 소프트웨어의 상기 업데이트된 버전을 설치하도록 구성됨 -,

상기 장치에 상기 소프트웨어의 상기 업데이트된 버전이 설치되었다는 표시를 수신하도록 하고,

상기 장치에 설치된 상기 소프트웨어 업데이트 버전 및 장치 정보에 대한 정보를 포함하는 블록을 생성하도록 하고,

상기 블록의 수락을 결정하고 상기 블록이 수락되는 경우 상기 블록을 상기 블록체인에 추가하는 피어 노드들의 상기 네트워크로 상기 블록을 전송하도록 하는,

시스템.

청구항 13

제12항에 있어서,

상기 메모리 장치는,

상기 적어도 하나의 프로세서에 의해 실행되는 경우, 상기 시스템으로 하여금,

상기 장치에 대한 상기 소프트웨어의 상기 업데이트된 버전을 식별하도록 하고,

상기 블록체인에 포함된 소프트웨어 인증 정보를 사용하여 상기 소프트웨어의 상기 업데이트된 버전을 인증하도록 하는

명령어들을 더 포함하는,

시스템.

청구항 14

제12항에 있어서,

상기 메모리 장치는,

상기 적어도 하나의 프로세서에 의해 실행되는 경우, 상기 시스템으로 하여금,

상기 블록체인의 사본을 저장하는 피어 노드로부터 상기 장치 이력들을 획득하도록 하는

명령어들을 더 포함하는,

시스템.

청구항 15

제12항에 있어서,

상기 메모리 장치는,

상기 적어도 하나의 프로세서에 의해 실행되는 경우, 상기 시스템으로 하여금,

피어 노드로부터 상기 블록체인의 일부를 획득하도록 하고,

상기 블록체인의 상기 일부로부터 상기 장치 이력들을 획득하도록 하는

명령어들을 더 포함하는,

시스템.

청구항 16

제12항에 있어서,

피어 노드들의 상기 네트워크에 의한 상기 블록의 수락은,

상기 블록을 수신한 과반수의 피어 노드들이 상기 블록을 수락한다는 동의를 통신했다는 결정에, 부분적으로, 기초하는,

시스템.

청구항 17

비-일시적인 기계 판독 가능한 저장 매체에 있어서,

하나 이상의 프로세서에 의해 실행되는 경우, 제1항 내지 제11항 중 어느 한 항의 방법을 수행하는 컴퓨터 프로그램을 저장하는,

비-일시적인 기계 판독 가능한 저장 매체.

청구항 18

삭제

청구항 19

삭제

청구항 20

삭제

발명의 설명

기 술 분 야

배 경 기 술

오늘날의 장치들은 종종 소프트웨어 업데이트를 통해 업데이트할 수 있는 소프트웨어를 실행하도록 구성된다. 예를 들어, 소프트웨어 업데이트들은 보안 문제들(security issues)이 발생한 경우, 해결(address)하고, 소프트웨어에서 발견된 오류들을 해결하고, 하드웨어 및/또는 주변 장치들(peripherals)의 동작을 개선하고, 새로운 장비 모델들(new models of equipment)에 대한 지원을 추가하기 위해 출시(release)될 수 있다. 이러한 업데이트들은 소프트웨어의 동작을 개선하고 하드웨어의 동작을 개선하며 소프트웨어 및 장치에 새로운 기능들을 추가할 수 있다. 예를 들어, 펌웨어 업데이트(firmware update)는 푸시 프로토콜(push protocol)을 사용하여 장치에 전송(sent)될 수 있으며, 장치는 펌웨어 업데이트 수신에 대한 응답으로 장치에 업데이트된 펌웨어를 설치하도록 구성될 수 있다.

블록체인 기술 또는 간단히 블록체인(blockchain)은 트랜잭션들의 목록들(lists of transactions)과 같은 정보를 저장하기 위해 사용할 수 있는 데이터 구조이다. 트랜잭션들은 블록들로 묶일(bundle) 수 있으며 각 블록(첫 번째 블록 또는 제네시스 블록(genesis block) 제외)은 해싱(hashing)을 사용하여 블록체인의 이전 블록을 다시 참조하거나 연결된다. 컴퓨터 노드(Computer nodes)들은 블록체인을 유지(maintain)하고 검증 프로세스(validation process)를 사용하여 블록체인에 추가된 각각의 새 블록을 암호학적으로(cryptographically) 검증(validate)한다. 검증 프로세스는, 독립적으로 입증될 수 있고 때때로 "작업 증명(proof-of-work)"이라고 불리는 계산적으로(computationally) 어려운 문제를 해결하는 것을 포함할 수 있다.

블록체인의 각 블록이 이전 블록의 암호화 해시 값(cryptographic hash value)을 참조하거나 포함하기 때문에 블록체인의 데이터 무결성(data integrity)(예: 이전에 기록된 트랜잭션이 수정되지 않았음을 확신)을 유지할 수 있다. 따라서, 블록이 이전 블록을 참조하면, 데이터를 조금만 수정해도 전체 블록의 해시 값에 영향을 미치기 때문에 이전 블록에 포함된 데이터(예: 트랜잭션들)를 수정하거나 변조(tamper)하기가 어려워진다. 블록

체인에 추가되는 각 블록은 앞선 블록(earlier block)의 내용(content)들을 변조하는 어려움을 증가시킨다.

발명의 내용

도면의 간단한 설명

도 1은 블록체인을 사용하여 소프트웨어 버전 이력을 관리하기 위한 예시적인 시스템 및 방법을 도시하는 도면이다.

도 2a 및 도 2b는 블록체인을 사용하여 장치 소프트웨어를 업데이트하고 소프트웨어 버전 이력을 관리하기 위한 시스템에 포함된 예시적인 구성요소들을 도시하는 블록도들이다.

도 3은 소프트웨어 업데이트를 소프트웨어 업데이트 서비스에 제공하고 소프트웨어 업데이트를 위한 소프트웨어 인증 정보를 블록체인에 저장하기 위한 예시적인 시스템 및 방법을 도시하는 블록도이다.

도 4는 블록체인에 포함된 인증 정보를 사용하여 소프트웨어 업데이트를 검증하기 위한 예시적인 시스템 및 방법을 도시하는 블록도이다.

도 5는 장치 상의 소프트웨어를 업데이트하고 장치에 대한 업데이트 이력을 블록체인에 저장하기 위한 예시적인 방법을 예시하는 흐름도이다.

도 6은 블록체인을 사용하여 버전 이력을 관리하는 방법을 실행하기 위해 사용될 수 있는 컴퓨팅 장치의 예를 도시하는 블록도이다.

발명을 실시하기 위한 구체적인 내용

블록체인을 사용하여 장치의 소프트웨어 버전 이력을 관리하고 보호하는 기술에 대해 설명한다. 블록체인은 장치에 설치된 소프트웨어 버전들을 추적(track)하기 위해 사용될 수 있다. 장치들은 로봇 장치들(robotic devices), 드론 장치들(drone devices), IoT(Internet of Things)(사물 인터넷) 장치들, 홈 자동화 장치들(home automation devices), 제조 장치들(manufacturing devices), 임베디드 장치들(embedded devices) 및 소프트웨어 업데이트들(software updates)을 수신하도록 구성된 기타 유형들의 컴퓨팅 장치들을 포함할 수 있다. 소프트웨어 업데이트는 장치에 설치된 소프트웨어를 업데이트, 업그레이드, 수정 또는 대체(replace)하는 소프트웨어 버전일 수 있다. 일부 경우들에서, 소프트웨어 업데이트는 이전에 장치에 존재하지 않았던 새 소프트웨어를 장치에 설치할 수 있다. 소프트웨어 업데이트는 펌웨어, 운영 체제(operating system), 응용 프로그램(application), 보안 파일들, 드라이버 파일들(driver files), 라이브러리 파일들(library files), 서비스들 등을 포함할 수 있지만 이에 제한되지 않는다. 소프트웨어 버전 정보는 블록체인에 기록될 수 있고, 블록체인은 네트워크의 컴퓨팅 노드들 간에 분산될(distributed) 수 있다. 컴퓨팅 노드들의 네트워크는 블록체인을 유지하고 검증 프로세스를 사용하여 블록체인에 추가된 각각의 새로운 블록을 암호학적으로 검증하도록 구성될 수 있다.

기술의 일 예에서 소프트웨어 업데이트 서비스는 장치에 소프트웨어 업데이트들을 제공하고, 장치에 설치된 소프트웨어의 이력을 제공하는 블록체인에 기록들(records)을 추가하기 위해 사용될 수 있다. 블록체인에 업데이트 이력을 기록하는 것은 장치에 설치된 소프트웨어 및 소프트웨어 버전들의 감사 추적(audit trail)을 제공할 수 있다. 설계에 의해, 블록체인들은 기본적으로(inherently) 블록체인에 포함된 데이터의 수정에 대해 저항(resistant)이 있으므로 일단 트랜잭션이 블록체인에 추가되면, 블록체인의 분산 특성(distributed nature)으로 인해 감지되지 않고 트랜잭션의 기록이 편집되거나 삭제될 수 없다. 블록체인의 데이터 무결성(즉, 수정에 대한 저항)은 장치의 업데이트 이력이 수정되는 것으로부터 보호하고 업데이트 이력이 유효하다는 높은 수준의 보증(assurance)을 제공하기 위해 사용될 수 있다.

업데이트 이력은 소프트웨어 버전 정보, 장치 정보(device information), 사용자 정보, 설치 날짜 정보, 장치 상태 정보, 네트워크 정보 및 기타 정보를 포함할 수 있지만 이에 제한되지 않는다. 일 예에서 블록체인을 사용하여 개별 장치의 업데이트 이력을 유지할 수 있다. 또 다른 예에서 블록체인은 다수의 장치들에 대한 업데이트 이력들을 유지하기 위해 사용될 수 있다. 또한, 일 예에서 소프트웨어 업데이트에 대한 인증 정보(authentication information)는 블록체인에서 유지될 수 있다. 인증 정보는, 나중에 더 자세히 설명하는 것처럼, 소프트웨어 업데이트 서비스 또는 장치가 장치에 소프트웨어 업데이트를 설치하기 전에 소프트웨어 업데이트(예: 설치 패키지, 실행 파일(executable), 파일 등)를 인증하기 위해 사용될 수 있다. 소프트웨어 업데이트

에 대한 인증 정보는 블록체인의 데이터 무결성을 사용하여 보호될 수 있다.

과거에는 소프트웨어 업데이트 보안이 중앙에서 관리되었다. 예를 들어, 중앙에 위치한 데이터베이스들은 장치들에 대한 소프트웨어 업데이트들의 배포(distribution)를 추적하기 위해 사용되었다. 악의적인 행위자들(Bad actors)은 장치에 침투(infiltrate)하고 탈취(take over)하려는 시도로 이러한 데이터베이스들을 악용(exploit)할 수 있다. 예를 들어, 악의적인 행위자는 장치에 악성 소프트웨어(malicious software)를 전송하는 OTA(Over the Air) 업데이트를 트리거(trigger)하기 위해 불법적으로(illegally) 데이터베이스에 접근(access)하고 장치에 대한 소프트웨어 업데이트 이력을 변경하거나 손상시킬 수 있다. 본 기술은 장치에 대한 소프트웨어 업데이트 정보를 블록체인에 기록함으로써 소프트웨어 업데이트 보안을 향상시키고, 컴퓨팅 노드들의 네트워크에 분산된 블록체인의 데이터 무결성을 사용하여 소프트웨어 업데이트 이력을 보호한다. 예를 들어 소프트웨어 업데이트 이력은 장치에 과거 버전들(past versions) 및 현재 설치된 소프트웨어의 버전에 대한 정보를 포함할 수 있다. 소프트웨어 업데이트 이력은 장치에 소프트웨어 업데이트를 전송하기 전에 장치에 설치된 소프트웨어 버전을 검증하기 위해 사용될 수 있다(예: 장치의 승인된(authorized) 소프트웨어의 버전이 소프트웨어 업데이트를 손상시키는 것을 방지하기 위해). 소프트웨어의 버전에 대한 소프트웨어 서명(software signature)과 같은, 소프트웨어 인증 정보(software authentication information)는 블록체인에 유지될 수 있고, 소프트웨어 인증 정보는 소프트웨어 업데이트를 장치로 전송하기 전에 소프트웨어 업데이트를 검증하기 위해 사용될 수 있다(예: 서명을 사용하여 소스(source)를 검증하는 것에 의해). 장치가 소프트웨어의 새 버전으로 업데이트된 후, 소프트웨어 업데이트와 연관된 정보가 블록체인에 기록될 수 있다. 결과적으로, 위에서 설명한 바와 같이, 본 기술은 블록체인의 데이터 무결성을 사용하여 장치에 대한 소프트웨어 업데이트들과 연관된 정보를 보호(securing)하는 개선들을 제공한다.

본 기술을 추가로 설명하기 위해 이제 도면들을 참조하여 예들이 제공된다. 도 1은 블록체인(112)을 사용하여 소프트웨어 버전 이력을 관리하기 위한 시스템(100) 및 방법의 높은 수준의 예를 도시하는 도면이다. 시스템(100)은 장치(102)(예를 들어, 로봇 장치) 및 블록체인(112)을 유지하도록 구성된 컴퓨팅 노드들(108)의 네트워크와 네트워크 통신하는 소프트웨어 업데이트 서비스(106)를 포함할 수 있다. 장치(102)는 소프트웨어 업데이트를 통해 업데이트 가능한 소프트웨어를 실행하도록 구성된 임의의 장치를 포함할 수 있다. 장치(102)의 비제한적인 예들은 로봇 장치, 드론 장치, IoT 장치, 홈 자동화 장치, 제조 장치, 모바일 장치(예를 들어, 스마트폰, 태블릿, e-리더(e-reader), 랩탑 컴퓨터 등), 데스크탑 컴퓨터, 서버 등을 포함한다. 블록체인(112)은 장치(102)에 대한 장치 정보, 장치(102)에 대한 소프트웨어 업데이트 이력, 및/또는 장치(102)에 설치가 가능할 수 있는 소프트웨어 업데이트에 대한 소프트웨어 인증 정보를 포함할 수 있지만 이에 제한되지 않는다. 컴퓨팅 노드들(108)의 네트워크에서 컴퓨팅 노드들(110)(또는 피어들(peers))은 블록체인(112)을 호스팅하고 컴퓨팅 노드들(110)은 (예를 들어, 인터넷, LAN(Local Area Network), WAN(Wide Area Network) 등을 사용하여) 서로 네트워크 통신한다. 컴퓨팅 노드들(110)은 컴퓨팅 노드들(112)의 네트워크와 소프트웨어 업데이트 서비스(106)가 블록체인(112)에 접근하고 수정하기 위해 서로 상호작용(interact)할 수 있도록 하는 API(Application Programming Interface)(응용 프로그래밍 인터페이스)들의 세트를 노출할 수 있다.

소프트웨어 업데이트 서비스(106)는 소프트웨어 업데이트들을 장치들(102)에 배포하도록 구성될 수 있다. 나중에 더 자세히 설명하는 바와 같이, 일 예에서, 소프트웨어 업데이트 서비스(106)는 블록체인(112)에 기록된 장치(102)에 대한 장치 정보를 검증할 수 있고(예를 들어, 이것이 유효한 장치인지 또는 허가된 장치(licensed device)인지 결정), 소프트웨어 업데이트 서비스(106)는 장치(102)에 소프트웨어 업데이트를 설치하도록 구성된 장치(102)에 소프트웨어 업데이트를 전송할 수 있다. 예를 들어, 소프트웨어 업데이트 서비스(106)는 장치(102)에 대한 장치 정보를 수신할 수 있다(예를 들어, 장치 정보 쿼리(query)를 통해 또는 장치(102)로부터 직접). 장치 정보는 장치 식별자(device identifier), 장치에 설치된 소프트웨어의 현재 버전 및 기타 정보를 포함할 수 있다. 소프트웨어 서비스(106)는 장치(106)의 아이덴터티 및 장치(106)에 설치된 소프트웨어의 버전을 확인하기 위해 장치 정보를 검증할 수 있다. 일 예에서, 장치 정보(예를 들어, 장치 식별자 및 소프트웨어 버전)는 블록체인(112)에 포함된 장치 기록에 대하여 검증될 수 있다.

장치 정보의 성공적인 검증(successful validation) 후에, 소프트웨어 업데이트 서비스(106)는 장치(102)에 대한 소프트웨어 업데이트를 식별(identify)하고 소프트웨어 업데이트를 장치(102)에 전송할 수 있다. 예를 들어, 소프트웨어 업데이트 서비스(106)는 장치(102)에 설치된 소프트웨어의 현재 버전(예를 들어, 버전 1.0)에 대한 정보를 사용하여 소프트웨어 업데이트(예를 들어, 버전 2.0)를 식별하고 소프트웨어 업데이트를 장치(102)에 전송할 수 있다. 일 예에서, 소프트웨어 업데이트를 장치(102)로 전송하기 전에, 소프트웨어 업데이트는 블록체인(112)에 포함된 소프트웨어 인증 정보(예를 들어, 소프트웨어 서명 또는 인증서)를 사용하여 인증될 수

있다. 후술하는 바와 같이, 소프트웨어 인증 정보는 블록체인(112)의 데이터 무결성을 사용하여 소프트웨어 인증 정보를 보호하기 위해 블록체인(112)에 저장될 수 있다.

소프트웨어 업데이트를 수신하는 것에 응답하여, 장치(102)는 장치(102)에 소프트웨어 업데이트를 설치한다. 소프트웨어 업데이트가 장치(102)에 설치된 후, 소프트웨어 업데이트 서비스(106)는 장치(102)에 대한 장치 식별자, 장치(102)에 설치된 소프트웨어 버전, 설치 날짜 등과 같은 소프트웨어 업데이트 정보를 포함하는 블록의 생성(generation of a block)을 개시(initiate)할 수 있다. 일 예에서, 소프트웨어 업데이트 서비스(106)는 소프트웨어 업데이트 정보를 노드들(110)에 전송할 수 있고, 노드들(110)은 블록을 생성 및 검증할 수 있고 블록을 블록체인(112)에 추가(append)할 수 있다. 또 다른 예에서, 소프트웨어 업데이트 서비스(106)는 소프트웨어 업데이트 정보를 포함하는 블록을 생성할 수 있고, 블록을 노드들(110)에 전송할 수 있고, 노드들(110)은 블록의 수락(acceptance)을 결정하고 블록을 블록체인(112)에 추가할 수 있다. 블록체인(112)에 블록을 추가할 때, 블록은 장치(102)에 대한 소프트웨어 업데이트 이력을 저장할 수 있고, 업데이트 이력은 블록체인(112)의 데이터 무결성 및 블록체인(112)의 분산된 특성을 통해 보호된다.

일 예에서, 블록체인(112)에 추가할 새로운 블록을 생성하는 것은 블록체인(112)에 추가된 최신 블록(latest block)(즉, 블록체인(112)에 추가된 마지막 블록)을 식별하고 마지막 블록으로부터 블록 식별자를 획득(obtain)하는 것을 포함할 수 있다. 마지막 블록의 블록 식별자는 소프트웨어 업데이트 정보와 함께 새 블록에 포함될 수 있고, 새로운 블록에 대한 블록 식별자는 새로운 블록에 포함된 정보에 해시 함수(hash function)를 적용하여 생성될 수 있다. 예를 들어, 새로운 블록에 대한 블록 식별자는 마지막 블록의 블록 식별자와 소프트웨어 업데이트 서비스(106)로부터 수신된 소프트웨어 업데이트 정보에 해시 함수를 적용하여 생성될 수 있다.

노드들(110)은 블록에 포함된 정보를 검증함으로써 블록의 수락을 결정할 수 있다. 예를 들어, 노드(110)는 원래 블록 식별자를 생성하기 위해 사용된 해싱 방법(hashing method)으로 블록 식별자를 재생하는 것(reproducing) 및 재생된 블록 식별자를 소프트웨어 업데이트 서비스(106)로부터 수신된 블록의 블록 식별자와 비교하는 것에 의해 소프트웨어 업데이트 서비스(106)에서 수신된 블록에 포함된 블록 식별자를 검증할 수 있다. 예로서, 노드(110)는 소프트웨어 업데이트 서비스(106)로부터 수신한 블록에 포함된 소프트웨어 업데이트 정보와 노드(110)에 저장된 블록체인(122)의 이전 블록으로부터 획득된 블록 식별자에 해시 함수를 적용하여 블록 식별자를 재생할 수 있다. 재생된 블록 식별자가 소프트웨어 업데이트 서비스(106)로부터 수신된 블록의 블록 식별자에 대응하는 경우, 블록은 검증된 것으로 간주되고 노드(110)는 노드(110)에 저장된 블록체인(112)의 사본(copy)에 블록을 추가할 수 있다. 재생된 블록 식별자가 소프트웨어 업데이트 서비스(106)로부터 수신된 블록의 블록 식별자와 일치하지 않는 경우, 블록은 블록체인(112)에 추가되지 않을 수 있다.

일 예에서, 노드들(110)은 어느 노드(110)가 블록체인(112)에 블록을 추가하는지를 결정하기 위해 작업 증명 처리를 수행(perform)할 수 있다. 예시로서, 소프트웨어 업데이트 서비스(106)는 소프트웨어 업데이트 트랜잭션들을 노드들(110)에 발행(publish)할 수 있고, 지연 속도(latency)로 인해 노드들(110) 소프트웨어 업데이트 트랜잭션들을 다양한 시간들에 수신할 수 있다. 그 결과, 노드들(110)에 의해 생성된 블록들은 다양한 소프트웨어 업데이트 트랜잭션들을 포함할 수 있고, 따라서 어느 노드(110)가 블록체인(112)에 블록을 추가하는지가 결정될 수 있다. 노드들(110)은 어느 노드(110)가 블록체인(112)에 블록을 추가하는지 결정하기 위해 작업 증명 처리를 수행할 수 있다. 예시적으로, 작업 증명 처리는 입증될 수 있는 컴퓨팅 계산(예를 들어, 암호 퍼즐)을 푸는 것(solving)을 포함할 수 있다. 작업 증명 처리를 완료하는 제1 노드(110)는 블록을 블록체인(112)에 추가하도록 선택된다. 일부 경우들에서, 블록체인(112)에 블록을 추가하도록 선택된 노드(110)는, 보상(award)을 받거나 다른 노드들(110)과 경쟁하기 위한 다른 유형의 인센티브(incentive)를 받을 수 있다. 보상(award)은 금전적인 보상일 수 있다. 다른 예에서, 지분 증명(proof-of stake)은 어느 노드(110)가 블록체인(112)에 블록을 추가하는지를 결정하기 위해 사용될 수 있다. 지분 증명은 결정론적 방법(deterministic method)을 사용하여 블록을 생성할 노드(110)를 선택하고, 노드(110)는 노드(110)가 유효한 블록을 생성하지 않는 경우 노드(110)가 잃을 수 있는 지분(stake)(예: 금전적 가치(monetary value))을 올린다(put up). 이해되는 바와 같이, 기술은 위에서 설명된 블록체인(112)에 블록을 생성하고 추가하기 위한 기술들로 제한되지 않는다. 블록을 생성하고 블록체인에 추가하기 위한 모든 기술이 고려되고 본 개시의 범위 내에 있다.

도 2a는 본 기술이 실행될 수 있는 예시적인 시스템 환경(system environment)(200)의 구성요소들을 도시한다. 시스템 환경(200)은 소프트웨어 업데이트 서비스(208) 및 장치(222) 및 소프트웨어 업데이트들(220)(예를 들어, 소프트웨어 실행 파일들의 버전들)에 대한 장치 이력들(device histories)(218)을 포함하는 데이터 저장소들(stores)(226a-b)과 소프트웨어 업데이트 서비스(208)를 호스팅하도록 구성된 복수의 서버들(206)을 포함할 수 있다. 시스템(200)은 하나 이상의 블록체인들(204)을 유지하기 위해 사용되는 복수의 노드들(202)을 포함할 수

있다. 노드(202)는 블록체인(204)에 저장된 정보를 검증, 노드(202)가 블록(224a-n)을 생성하는지 여부를 결정하기 위해 작업 증명 처리를 수행, 장치(222) 및/또는 소프트웨어 업데이트와 관련된 정보를 포함하는 블록(224a-n)을 생성 및 블록(224a-n)을 블록체인(204)에 추가하는 것과 같은 블록체인 기능들을 수행하기 위해 사용되는 블록체인 소프트웨어를 실행하도록 구성된 서버 또는 다른 컴퓨팅 장치를 포함할 수 있다.

블록체인(204)은 장치(222)에 설치된 소프트웨어의 장치 이력을 포함하는 복수의 연결된 블록들(224a-n)을 포함할 수 있다. 블록체인(204)은 단일 장치(single device)(222)에 대한 장치 이력 또는 다수의 장치들(multiple devices)(222)에 대한 장치 이력들을 포함할 수 있다. 예를 들어, 블록체인(204)은 IoT 장치들의 네트워크, 로봇 장치들의 네트워크, 드론 장치들의 네트워크 등과 같은 장치 네트워크에 포함된 장치들(222)에 대한 장치 이력들을 포함할 수 있다. 장치(222)에 대한 장치 이력은 장치 식별자(예: 고유 장치 식별자(unique device identifier)(UDI), 범용 고유 식별자(Universal Unique Identifier)(UUID), MAC 주소(Media Access Control Address), 국제 모바일 장비 아이덴터티(International Mobile Equipment Identity)(IMEI), 독점 고유 하드웨어 식별자(proprietary unique hardware identifier) 등), 과거 장치 정보(historical device information)(예: 마지막으로 알려진 위치, 네트워크 주소, 장치 상태 등), 소프트웨어 버전 정보(예: 소프트웨어 버전 번호, 보안 패치 수준 및 날짜, 소프트웨어 빌드 번호(software build number) 등) 및 장치(222)에 대한 장치 이력이 블록체인(204)에 추가된 때를 나타내는 시간 기준(time reference)을 포함할 수 있다. 일 예에서, 블록(224n)에 도시된 바와 같이, 블록체인(204)은 후술되는 바와 같이 소프트웨어 업데이트(220)를 검증하기 위해 사용되는 소프트웨어 인증 정보를 포함할 수 있다.

일 예에서, 블록체인(204)에 포함된 장치 이력은 소프트웨어 업데이트 서비스(208)에 접근 가능한 장치 이력 데이터 저장소(226a)에 저장될 수 있다. 예를 들어, 블록체인(204)의 사본은 노드(202)로부터 획득될 수 있고 장치 이력은 블록체인(204)으로부터 추출되어 장치 이력 데이터 저장소(226a)에 저장될 수 있다. 장치 이력 데이터 저장소(226a)에 포함된 장치 이력(218)은 장치 이력 데이터 저장소(226a)에 포함된 장치 이력(218)의 데이터 무결성을 유지하기 위해 블록체인(204)으로부터 정기적으로 업데이트될 수 있다. 다른 예에서, 장치 이력은 노드(202)로부터 블록체인(204) 또는 블록체인(204)의 일부(portion)를 검색하는 것(retrieving)에 의해 블록체인(204)으로부터 직접 획득될 수 있다.

소프트웨어 업데이트 서비스(208)는 소프트웨어 업데이트 서비스(208)에 등록된 장치들(222)에 소프트웨어 업데이트들(220)을 제공할 수 있다. 일 예에서, 소프트웨어 업데이트 서비스(208)는 장치 인증 모듈(210), 소프트웨어 인증 모듈(212), 소프트웨어 배포 모듈(software distribution module)(214), 및 블록체인 모듈(216)을 포함하지만 이에 제한되지 않는 다수의 모듈들을 포함할 수 있다. 장치 인증 모듈(210)은 장치(222)의 아이덴터티(identity)를 확인(confirm)하기 위해 장치(222)를 인증하도록 구성될 수 있다. 장치(222)는 장치(222)로부터 수신된 장치 정보를 블록체인(204)에 포함된 장치 정보와 비교함으로써 인증될 수 있다. 예시적으로, 장치(222)에 대한 장치 정보는 장치 식별자 및/또는 장치 특성들 (예를 들어, 장치 유형, 장치 계정, IP 주소, 장치(222)에 설치된 소프트웨어의 현재 버전 등)을 포함할 수 있고, 이는 블록체인(204)에 포함된 장치 이력(218)으로부터 획득된 장치 정보에 대해 검증될 수 있다. 비제한적인 예로서, 장치(222)로부터 수신된 MAC 주소, 장치 계정 번호 및 장치 모델 번호는 장치 이력 (218)에 포함된 MAC 주소, 장치 계정 번호 및 장치 모델 번호에 대해 검증될 수 있다. 장치 이력(218)은 장치 이력 데이터 저장소(226a) 및/또는 블록체인(204)으로부터 획득될 수 있다.

소프트웨어 배포 모듈(214)은 장치(222)가 소프트웨어 업데이트(220)를 수신하고 장치(222)에 소프트웨어 업데이트(220)의 전송을 개시할 자격이 있는지(eligible) 여부를 결정하도록 구성될 수 있다. 예시적으로, 소프트웨어 업데이트(220)는 장치(222)로부터 소프트웨어 업데이트 요청을 수신하는 것에 응답하여 (예를 들어, 장치(222)는 소프트웨어 업데이트(220)가 가능한지 여부를 결정하기 위해 소프트웨어 업데이트 서비스(208)와 주기적으로 통신하고 "체크-인(check-in)"할 수 있다), 및/또는 장치(222)가 소프트웨어 업데이트(220)에 자격이 있다는 결정에 응답하여 (예를 들어, 장치 이력들(218)은 소프트웨어 업데이트(220)를 수신할 자격이 있는(eligible) 장치들(222)을 식별하기 위해 쿼리될 수 있다) 장치(222)에 전송될 수 있다.

장치(222)로부터 소프트웨어 업데이트 요청을 수신하는 것에 응답하여, 소프트웨어 배포 모듈(214)은 소프트웨어 업데이트(220)가 장치(222)에 대해 가능한지를 결정하기 위해 사용될 수 있다. 예시로, 장치(222)에 대한 장치 식별자를 사용하여, 소프트웨어 배포 모듈(214)은 장치(222)에 현재 설치된 소프트웨어의 버전을 식별하기 위해 장치(222)와 연관된 (예를 들어, 장치 이력 데이터 저장소(226a)에 위치되고/거나 블록체인(204)으로부터 직접 획득된) 장치 이력(device history)(218)을 참조(reference)할 수 있고, 소프트웨어 배포 모듈(214)은 소프트웨어 업데이트 데이터 저장소(226b)에 쿼리(query)하여 소프트웨어의 새 버전이 장치(222)에 설치될 수 있

는지 여부를 결정할 수 있다. 소프트웨어 업데이트(220)가 가능한 경우, 소프트웨어 배포 모듈(214)은 장치(222)에 설치하기 위해 장치(222)에 소프트웨어 업데이트(220)의 전송을 개시할 수 있다.

위에서 언급한 바와 같이, 소프트웨어 배포 모듈(214)은 소프트웨어 업데이트(220)를 수신할 자격이 있는 장치들(222)을 식별하기 위해 사용될 수 있다. 예를 들어, 소프트웨어 업데이트(220)가 출시될 수 있고 소프트웨어 배포 모듈은 새로 출시된 소프트웨어 업데이트(220)를 수신할 자격이 있는 장치들(222)을 식별하기 위해 사용될 수 있다. 소프트웨어 배포 모듈(214)은 장치(222)와 연관된 장치 이력(218)를 참조함으로써 장치(222)가 소프트웨어 업데이트(220)를 수신할 자격이 있는지 여부를 결정할 수 있고, 장치(222)에 현재 설치된 소프트웨어 버전, 장치 유형, 장치 위치(예를 들어, 네트워크 위치, 지리적 위치 등), 현재 장치 사용, 및 기타 요인들(factors)과 같은 요인들에 기초하여 장치(222)에 대해 소프트웨어 업데이트(220)가 가능한지 여부를 결정할 수 있다.

일례에서, 소프트웨어 배포 모듈(214)은 장치(222)에 현재 설치된 소프트웨어 버전이 장치 이력(218)에 기록된 소프트웨어 버전에 대응하는지 입증하도록 구성될 수 있다. 예를 들어, 장치(222)로부터 수신된 장치 정보는 장치(222)에 설치된 소프트웨어의 버전에 대한 정보를 포함할 수 있고, 소프트웨어 배포 모듈(214)은 장치(222)로부터 수신된 소프트웨어 버전 정보를 장치(222)에 현재 설치된 장치 이력(218)에 기록된 소프트웨어 버전과 비교(compare)할 수 있다. 소프트웨어 버전들 간의 불일치(disagreement)는 장치(222)가 손상되(compromised)었으며 장치(222)를 복구하기 위해 적절한 조치가 취해질 수 있음을 나타낼 수 있다. 장치(222)에 현재 설치된 소프트웨어 버전이 장치 이력(218)에 기록된 소프트웨어 버전에 대응하는 경우, 소프트웨어 배포 모듈(214)은 장치(222)로의 소프트웨어 업데이트(220)의 전송을 개시할 수 있다.

소프트웨어 배포 모듈(214)은 소프트웨어 업데이트 데이터 저장소(226b)에서 소프트웨어 업데이트(220)(예를 들어, 소프트웨어의 사본)를 획득하고 네트워크(228)를 통해 장치(222)에 소프트웨어 업데이트(220)를 전송함으로써 장치(222)에 소프트웨어 업데이트(220)를 전송하거나 또는 소프트웨어 업데이트 데이터 저장소(226b)가 장치(222)에 소프트웨어 업데이트(220)를 전송하도록 명령하는 것을 개시할 수 있다. 소프트웨어 업데이트(220)는 장치(222)에 소프트웨어 업데이트(220)를 설치하라는 명령어들(instructions)과 함께 장치(222)로 전송될 수 있다. 또한, 소프트웨어 배포 모듈(214)은 네트워크 위치로부터 소프트웨어 업데이트(220)를 획득하고 장치(222)에 소프트웨어 업데이트(220)를 설치하라는 명령어들과 함께 소프트웨어 업데이트(220)의 네트워크 위치(예를 들어, 인터넷 식별자(uniform Resource Identifier; URI))를 장치(222)에 전송할 수 있다. 장치(222)에 소프트웨어 업데이트(220)를 설치한 후, 장치(222)는 소프트웨어 업데이트(220)가 장치(222)에 성공적으로 설치되었음을 나타내는 메시지를 소프트웨어 업데이트 서비스(106)에 전송할 수 있다.

소프트웨어 인증 모듈(212)은 블록체인(204)에 포함된 소프트웨어 서명을 사용하여 소프트웨어 업데이트(220)를 검증하도록 구성될 수 있다. 예를 들어, 소프트웨어 인증 모듈(212)은 (예를 들어, 개인 키(private key)로 소프트웨어 업데이트(220)를 암호화(encrypting)하거나 소프트웨어 업데이트(220)에 해시 함수를 적용함으로써) 소프트웨어 업데이트(220)에 대한 소프트웨어 서명을 생성하기 위해 사용될 수 있고, 소프트웨어 서명은 블록체인(204)에 저장될 수 있다(블록(224n)에 도시됨). 블록체인(204)에 소프트웨어 서명을 저장하면 블록체인(204)의 데이터 무결성 및 분산 특성을 사용하여 소프트웨어 서명이 변경되는 것을 방지할 수 있다. 이와 같이, 소프트웨어 서명을 수정하거나 교체하려는 모든 시도는 블록체인(204)에 변경이 이루어졌는지 여부를 결정하기 위해 블록체인(204)을 참조함으로써 감지될 수 있다.

일 예에서, 소프트웨어 인증 모듈(212)은 소프트웨어 업데이트(220)에 대한 소프트웨어 서명을 (예: 블록체인(204)으로부터) 획득하고 소프트웨어 업데이트(220)에 해시 함수를 적용하고 해시 함수의 결과들을 소프트웨어 업데이트(220)와 연관된 소프트웨어 서명과 비교함으로써 소프트웨어 업데이트(220)를 검증할 수 있다. 소프트웨어 업데이트(220)의 성공적인 검증 후에, 소프트웨어 업데이트(220)는 하나 이상의 장치들(222)로 전송될 수 있다. 다른 예에서, 소프트웨어 인증 모듈(212)은 소프트웨어 서명을 장치(222)에 제공할 수 있고 장치(222)는 소프트웨어 서명을 사용하여 소프트웨어 업데이트(220)를 검증할 수 있다.

블록체인 모듈(216)은 블록체인(204)과 연관된 기능을 수행하도록 구성될 수 있다. 일 예에서, 블록체인 모듈(216)은 블록(224a-n)을 생성하고 블록들(224a-n)을 블록체인(204)에 추가하는 노드들(202)에 블록(224a-n)을 전송함으로써 블록(224a-n)을 블록체인(204)에 추가하기 위해 사용될 수 있다. 다른 예에서, 블록체인 모듈(216)은 정보를 포함하고 블록(224a-n)을 블록체인(204)에 추가하기 위해 하나 이상의 노드들(202)이 블록(224a-n)을 생성할 수 있게 하는 노드들(202)의 네트워크에서 노드들(202)에 정보(예: 소프트웨어 업데이트 정보 및/또는 소프트웨어 서명들)를 전송하기 위해 사용될 수 있다. 또 다른 예에서, 블록체인 모듈(216)은 블록

체인(204) 또는 블록체인(204)의 일부를 획득하고 블록체인(204)에 저장된 장치 이력 정보를 임의의 모듈들(210/212/214)에 제공하기 위해 사용될 수 있다. 또한, 일 예에서, 블록체인 모듈(216)은 블록체인(204)으로부터 장치 이력 정보를 얻고(예를 들어, 주기적으로 또는 온디맨드로(on demand)) 장치 이력 정보를 모듈들(210/212/214)에 접근할 수 있는 장치 이력 데이터 저장소(226a)에 저장하기 위해 사용될 수 있다.

시스템 환경(200)은 물리적 머신(physical machine)을 에뮬레이트(emulate)하도록 구성된 머신(즉, 컴퓨터)의 소프트웨어 구현의 경우일 수 있는 가상 머신들(virtual machines)을 실행하기 위한 컴퓨팅 리소스들을 포함할 수 있다. 일 예에서, 서버(206)는 소프트웨어 업데이트 서비스(208)를 호스팅하기 위해 사용되는 가상 머신을 포함할 수 있다.

대안적인 예에서, 도 2b에 도시된 바와 같이, 소프트웨어 업데이트 서비스(208)의 하나 이상의 모듈들은 장치(222)에서 호스팅될 수 있고 모듈들은 위에서 설명된 바와 같이 장치(222)에 설치된 소프트웨어를 업데이트하기 위해 사용될 수 있다. 예를 들어, 소프트웨어 업데이트 서비스(208)는 장치(222) 상에서 실행될 수 있고, 소프트웨어 업데이트 서비스(208)는 블록체인(204)에 포함된 장치 이력으로 장치(222)에 저장된 장치 정보를 검증하고 업데이트 데이터 저장소(226b)로부터 소프트웨어 업데이트(220)를 검색함으로써 장치(222)에 대한 소프트웨어 업데이트를 개시할 수 있다. 소프트웨어 업데이트가 장치(222)에 설치된 후, 소프트웨어 업데이트 서비스(208)는 장치에 설치된 소프트웨어 업데이트에 대한 정보를 포함하도록 블록 생성을 개시하고 위에서 설명된 방법들 중 하나를 사용하여 블록을 블록체인(204)에 추가할 수 있다.

시스템 환경(200) 내에 포함된 다양한 프로세스들 및/또는 다른 기능은 하나 이상의 메모리 모듈들과 통신하는 하나 이상의 프로세서들에서 실행될 수 있다. 시스템 환경(200)은, 예를 들어, 하나 이상의 서버 뱅크들(server banks) 또는 컴퓨터 뱅크들 또는 다른 배열들로 배열된 다수의 컴퓨팅 장치들을 포함할 수 있다. 컴퓨팅 장치들은 하이퍼바이저들(hypervisors), 가상 머신 모니터들(virtual machine monitors)(VMM) 및 기타 가상화 소프트웨어(virtualization software)를 사용하여 컴퓨팅 환경을 지원할 수 있다. "데이터 저장소"라는 용어는 데이터를 저장, 접근, 구성 및/또는 검색할 수 있는 모든 장치 또는 장치의 조합을 나타낼 수 있고, 이는 데이터 서버, 관계형 데이터베이스, 객체 지향 데이터베이스, 클러스터 저장 시스템, 데이터 저장 장치, 데이터 웨어하우스, 플랫 파일 및 중앙 집중식, 분산형 또는 클러스터형 환경의 데이터 저장 구성의 조합 및 수를 포함할 수 있다. 이는 데이터 저장소의 저장소 시스템 구성요소들에는 스토리지 에이리어 네트워크(Storage Area Network)(SAN), 클라우드 저장소 네트워크, 휘발성 또는 비휘발성 RAM, 광학 미디어 또는 하드 드라이브 유형 미디어와 같은 저장소 시스템들이 포함될 수 있다. 데이터 저장소는 이해될 수 있는 바와 같이 복수의 데이터 저장소들을 나타낼 수 있다.

시스템 환경(200)에 포함된 모듈 및 서비스와 관련하여 수행될 수 있는 API 호출, 프로시저 호출 또는 기타 네트워크 명령어는 웹 표현상태 변경(Representational State Transfer)(REST) 기술 또는 단순 객체 접근 프로토콜(Simple Object Access Protocol)(SOAP) 기술을 포함하되 이에 국한되지 않는 다양한 기술들에 따라 구현될 수 있다. REST는 분산 하이퍼미디어 시스템들(distributed hypermedia systems)을 위한 아키텍처 스타일(architectural style)이다. RESTful API(RESTful 웹 서비스라고도 함)는 HTTP 및 REST 기술을 사용하여 구현된 웹 서비스 API이다. SOAP는 웹 기반 서비스들의 컨텍스트(context)에서 정보를 교환하기 위한 프로토콜이다.

네트워크(228)는 인트라넷, 인터넷, 근거리 네트워크, 광역 네트워크, 무선 데이터 네트워크, 또는 임의의 다른 그러한 네트워크 또는 이들의 조합을 포함하는 임의의 유용한 컴퓨팅 네트워크를 포함할 수 있다. 그러한 시스템에 사용되는 구성요소들은 선택된 네트워크 및/또는 환경의 유형에 적어도 부분적으로 의존할 수 있다. 네트워크(228)를 통한 통신은 유선 또는 무선 연결 및 이들의 조합에 의해 활성화될 수 있다.

도 2a 및 2b는 특정 처리 모듈들이 이 기술과 관련하여 논의될 수 있고 이러한 처리 모듈들이 컴퓨팅 서비스들로서 구현될 수 있음을 예시한다. 하나의 예시적인 구성에서, 모듈은 서버 또는 다른 컴퓨터 하드웨어에서 실행되는 하나 이상의 프로세스들이 있는 서비스로 간주될 수 있다. 이러한 서비스들은 요청들을 수신하고 다른 서비스들 또는 소비자 장치들(consumer devices)에 출력을 제공할 수 있는 중앙에서 호스팅되는 기능 또는 서비스 응용 프로그램일 수 있다. 예를 들어, 서비스들을 제공하는 모듈들은 서버, 가상화된 서비스 환경, 그리드 또는 클러스터 컴퓨팅 시스템에서 호스팅되는 온디맨드 컴퓨팅으로 간주될 수 있다. API는 제2 모듈이 제1 모듈에 요청들을 전송하고 출력을 수신할 수 있도록 각 모듈에 제공될 수 있다. 이러한 API는 또한 제3자들이 모듈과 인터페이스하고 요청을 하고 모듈들로부터 출력을 수신하도록 허용할 수 있다. 도 2a 및 2b는 위의 기술들을 구현할 수 있는 시스템의 예들을 예시하지만, 많은 다른 유사하거나 다양한 환경들이 가능하다. 위에서

논의되고 예시된 예시적인 환경들은 단지 대표적인 것이며 제한적이지 않다.

도 3은 소프트웨어 업데이트(324)를 소프트웨어 업데이트 서비스(308)에 제공하고 소프트웨어 업데이트(324)에 대한 소프트웨어 인증 정보를 블록체인(310)에 저장하기 위한 예시적인 시스템(300) 및 방법을 도시하는 블록도이다. 소프트웨어 인증 정보는 소프트웨어 작성자 및/또는 소프트웨어 빌드 시스템의 아이덴터티를 입증하기 위해 사용되는 소프트웨어 서명(306), 및 소프트웨어 업데이트(324)의 컴퓨터 코드가 변경되거나 손상되지 않았는지 확인하기 위한 체크섬(checksum)을 포함할 수 있다. 인증 정보는 또한 소프트웨어 업데이트(324)에 대한 버전 정보 및 소프트웨어 업데이트(324)에 대한 기타 메타데이터를 포함할 수 있다. 단순함의 이점을 위해, 아래의 설명은 다른 또는 추가 소프트웨어 인증 정보가 사용될 수 있다는 이해와 함께 소프트웨어 서명(306)에 초점을 맞출 것이다.

도시된 바와 같이, 소프트웨어 업데이트(324)는 소프트웨어 업데이트 서비스(308)를 호스팅하는 서버(304)에 제공될 수 있다. 소프트웨어 업데이트(324)는 장치에 설치된 소프트웨어를 업데이트, 업그레이드 또는 대체하는 소프트웨어 버전일 수 있다. 관리자는 소프트웨어 업데이트(324)(예를 들어, 실행 파일)를 서버(304)에 업로드할 수 있고 소프트웨어 업데이트(324)는 소프트웨어 업데이트 서비스(308)에 액세스 가능한 데이터 저장소(320)에 저장될 수 있다.

일 예에서, 소프트웨어 서명(306)은 소프트웨어 업데이트(324)에 포함될 수 있다. 예를 들어, 소프트웨어 업데이트(324)는 공개/개인 키 쌍을 포함하는 소프트웨어 서명 기술을 사용하여 서명(예를 들어, 암호화)될 수 있다. 예를 들어, 소프트웨어 개발자는 소프트웨어 업데이트(324)를 빌드한 후 개인 키를 사용하여 소프트웨어 업데이트(324)에 서명할 수 있다. 서명은 개발자의 개인 키 또는 신뢰할 수 있는 인증 기관(CA)에서 얻은 개인 키를 사용하여 수행할 수 있고, 소프트웨어 업데이트(324)는 소프트웨어 서명(306)(예를 들어, 공개 키 기반 구조(public key infrastructure)(PKI) 인증서(certificate))와 함께 서버(304)에 업로드될 수 있다. 소프트웨어 업데이트(324) 및 소프트웨어 서명(306)이 서버(304)에 업로드된 후, 블록(312)은 소프트웨어 서명(306)(예를 들어, PKI 인증서) 및 소프트웨어 업데이트(324)에 대한 정보(예를 들어, 버전 번호와 같은 소프트웨어 업데이트 식별자)를 포함하도록 생성될 수 있다. 블록(312)은 블록체인(310)에 추가될 수 있다. 일 예에서, 소프트웨어 업데이트 서비스(308)는 블록(312)이 수락(accept)되는 경우에 소프트웨어 서명(306)을 포함하는 블록(312)을 생성하고 블록(312)의 수락을 결정하고 블록(312)을 블록체인(310)에 첨부하도록 구성된 노드(302)의 네트워크에 블록(312)을 전송할 수 있다. 다른 예에서, 소프트웨어 업데이트 서비스(308)는 소프트웨어 서명(306) 및 소프트웨어 업데이트(324)에 대한 정보를 노드들(302)의 네트워크로 전송할 수 있고, 노드들(302)은 블록(312)을 생성하고 블록(312)을 블록체인(310)에 추가할 수 있다.

다른 예에서, 소프트웨어 업데이트 서비스(308)는 암호화 해시 함수(예를 들어, MD5, SHA-1, SHA-256 등)를 사용하여 소프트웨어 업데이트(324)에 대한 소프트웨어 서명(306)을 생성하도록 구성될 수 있다. 예를 들어, 소프트웨어 업데이트(324)를 서버(304)에 업로드한 후, 소프트웨어 업데이트 서비스(308)는 소프트웨어 업데이트(324)에 암호화 해시 함수를 적용하여 소프트웨어 서명(306)을 생성할 수 있다. 그 다음, 소프트웨어 업데이트 서비스(308)는 일 예에서 소프트웨어 서명(306) 및 소프트웨어 업데이트(324)에 대한 정보를 노드들(302)의 네트워크로 전송하여 블록체인(310)의 블록(312)에 저장할 수 있다. 다른 예에서, 소프트웨어 업데이트 서비스(308)는 소프트웨어 서명(306) 및 소프트웨어 업데이트(324)에 대한 정보를 포함하는 블록(312)을 생성할 수 있고, 소프트웨어 업데이트 서비스(308)는 블록(312)을 블록체인(310)에 추가될 노드들(302)의 네트워크로 전송할 수 있다. 일 예에서, 블록체인(310)은 소프트웨어 업데이트를 위한 소프트웨어 서명들을 저장하는 것에 전념할 수 있다. 다른 예에서, 블록체인(310)은 소프트웨어 업데이트에 대한 소프트웨어 서명들, 앞서 설명한 장치들에 대한 업데이트 이력들, 그리고 소프트웨어 업데이트와 관련된 기타 정보(예: 소프트웨어 업데이트의 네트워크 저장 위치)를 모두 저장하기 위해 사용될 수 있다.

도 4는 블록체인(310)에 저장된 소프트웨어 서명(306)을 사용하여 소프트웨어 업데이트(324)를 검증하기 위한 시스템(300) 및 방법을 예시하는 블록도이다. 장치(322)에서 소프트웨어 업데이트의 일부로서, 소프트웨어 업데이트(324)가 변경되거나 손상되지 않았음을 보장하기 위해 블록체인(310)에 저장된 소프트웨어 서명(306)을 사용하여 소프트웨어 업데이트(324)가 검증될 수 있다.

일 예에서, 장치(322) 상의 소프트웨어 업데이트의 일부로서, 소프트웨어 업데이트(324)(예를 들어, 실행 파일)는 장치(322)에 대해 식별될 수 있고 소프트웨어 업데이트(324)는 소프트웨어 업데이트 데이터 저장소(320)로부터 검색될 수 있다. 일 예에서, 소프트웨어 업데이트(324)와 연관된 소프트웨어 서명(306)은 블록체인(310)의 블록(312)으로부터 검색될 수 있고, 소프트웨어 서명(306)은 소프트웨어 서명을 인증하기 위해 공개 키

를 사용하여 검증될 수 있다. 일 예에서 공개 키는 보안 PKI를 사용하여 신뢰할 수 있는 루트 인증 기관(certificate authority)(CA)으로 역추적(traceable back)될 수 있다. 일 예에서, 소프트웨어 서명(306)의 검증은 소프트웨어 업데이트(324)를 장치(322)로 전송하기 전에 소프트웨어 업데이트 서비스(308)에 의해 수행될 수 있다. 다른 예에서, 소프트웨어 서명(306)의 검증은 소프트웨어 업데이트(324)로 장치(322) 상의 소프트웨어 업데이트의 일부로서 장치(322)에 의해 수행될 수 있다.

다른 예에서, 소프트웨어 서명(306)은 블록체인(310)으로부터 검색될 수 있고 소프트웨어 서명(306)은 소프트웨어 업데이트 데이터 저장소(320)의 소프트웨어 업데이트(324)와 함께 저장된 소프트웨어 서명과 비교될 수 있다. 예를 들어, 소프트웨어 서명(306)은 암호화 해시 함수를 소프트웨어 업데이트 파일에 적용함으로써 생성될 수 있고, 결과 소프트웨어 서명(resulting software signature)(306)은 블록체인(310)에 저장될 수 있고, 소프트웨어 서명(306)의 사본은 소프트웨어 업데이트(324)와 함께 소프트웨어 업데이트 데이터 저장소(320)에 저장될 수 있다. 장치(322)가 소프트웨어 업데이트(324)를 수신할 때, 소프트웨어 업데이트(324)에 대한 소프트웨어 서명(306)은 블록체인(310)에서 소프트웨어 서명(306)을 검색하고 소프트웨어 업데이트 데이터 저장소(320)에서 소프트웨어 서명(306)의 사본을 검색함으로써 검증될 수 있고, 소프트웨어 서명들이 일치하는지 확인하기 위해 소프트웨어 서명들이 비교될 수 있다. 일 예에서, 소프트웨어 업데이트 서비스(308)는 소프트웨어 서명을 비교하기 위해 사용될 수 있다. 다른 예에서, 소프트웨어 서명(306) 및 소프트웨어 서명(306)의 사본은 소프트웨어 업데이트(324)와 함께 장치(322)로 전송될 수 있고, 장치(322)는 소프트웨어 서명(306)과 소프트웨어 서명(306)의 사본을 비교함으로써 소프트웨어 서명(306)을 검증할 수 있다.

도 5는 장치 상의 소프트웨어를 업데이트하고 장치에 대한 업데이트 이력을 블록체인에 저장하기 위한 예시적인 방법(500)을 예시하는 흐름도이다. 블록(510)에서와 같이, 소프트웨어 업데이트들을 수신하도록 구성된 장치에 대한 장치 정보가 수신될 수 있고, 장치에 대한 장치 이력은 다수의 네트워크 노드들(multiple network nodes) 사이에 분산된 블록체인에 포함될 수 있다. 예를 들어, 장치 정보는 서비스 제공자 환경(service provider environment)에서 호스팅되는 소프트웨어 업데이트 서비스 또는 장치 자체에서 호스팅되는 소프트웨어 업데이트 서비스에서 수신될 수 있다.

블록(520)에서와 같이, 장치 정보는 장치의 아이덴터티(identity of the device)를 확인하기 위해 블록체인에 포함된 장치 이력에 대해 검증될 수 있다. 일 예에서, 장치 이력에 대해 장치 정보를 검증하는 단계는, 블록체인으로부터 장치 이력을 획득하고, 장치의 아이덴터티를 확인하기 위하여, 장치 이력에 기록된 장치 식별자에 대하여 장치 정보에 포함된 장치 식별자를 검증하고, 장치에 현재 설치된 소프트웨어의 버전을 확인하기 위하여, 장치 이력에 기록된 소프트웨어 버전에 대하여 장치 정보에 포함된 소프트웨어 버전 식별자를 체크(check)하는 단계를 포함할 수 있다. 일 예에서, 블록체인 사본을 저장하는 피어 노드(peer node)에서 장치 이력을 획득할 수 있다. 다른 예에서, 블록체인의 일부는 피어 노드에서 얻을 수 있고 장치에 대한 장치 이력은 블록체인의 일부에서 획득될 수 있다.

블록(530)에서와 같이, 장치에 소프트웨어 업데이트를 설치(install)하도록 구성된 장치에 대한 소프트웨어 업데이트가 개시될 수 있다. 일 예에서, 소프트웨어 업데이트를 개시하는 단계는 소프트웨어 업데이트를 장치에 설치하라는 명령어들(instructions)과 함께 소프트웨어 업데이트와 연관된 소프트웨어 사본을 장치에 전송하는 단계를 포함할 수 있다. 다른 예에서, 소프트웨어 업데이트를 개시하는 단계는 네트워크 위치로부터 소프트웨어 사본을 획득하고 소프트웨어 사본을 장치에 설치하라는 명령어들과 함께 소프트웨어 사본의 네트워크 위치를 장치에 전송하는 단계를 포함할 수 있다. 또 다른 예에서, 장치는 소프트웨어 업데이트 서비스에서 소프트웨어 업데이트의 소프트웨어 사본에 대한 요청을 전송하거나 장치 자체의 원격 저장소 위치(remote storage location) 또는 로컬 저장소 위치(local storage location)에서 소프트웨어 업데이트의 소프트웨어 사본을 검색하여 소프트웨어 업데이트를 시작할 수 있다.

일 예에서, 소프트웨어 업데이트가 장치에 설치될 수 있음을 식별하는 일부로서, 블록체인에 포함된 소프트웨어 인증 정보를 사용하여 장치에 대한 소프트웨어의 업데이트된 버전이 식별되고 소프트웨어의 업데이트된 버전이 인증될 수 있다. 예를 들어, 소프트웨어 인증 정보(예: 소프트웨어 업데이트를 위한 소프트웨어 서명)는 블록체인에서 획득될 수 있고, 일 예에서, 소프트웨어 인증 정보는 소프트웨어 업데이트를 검증하기 위해 소프트웨어 업데이트 서비스에 의해 사용될 수 있거나, 다른 예에서 소프트웨어 인증 정보는 소프트웨어 인증 정보를 사용하여 소프트웨어 업데이트를 검증하도록 구성될 수 있는 장치에 제공될 수 있다.

블록(540)에서와 같이, 소프트웨어 업데이트가 장치에 설치되었다는 표시(indication)가 수신될 수 있다. 예를 들어, 장치는 소프트웨어 업데이트가 장치에 성공적으로 설치되었음을 나타내는 메시지를 소프트웨어 업데이트

서비스에 전송할 수 있다. 다른 예로서, 소프트웨어 업데이트 서비스는 소프트웨어 업데이트가 설치되었는지 여부를 결정하기 위해 주기적으로 장치를 쿼리할 수 있다.

블록(550)에서와 같이, 블록의 생성은 장치에 대한 정보 및 장치에 설치된 소프트웨어 업데이트뿐만 아니라 다른 정보를 포함하도록 개시될 수 있다. 일 예에서, 블록 생성을 개시하는 단계는, 소프트웨어 업데이트가 장치에 설치되었음을 나타내는 업데이트된 장치 정보를 다수의 네트워크 노드들에 전송하는 단계 및 어떤 네트워크 노드가 장치에 설치된 소프트웨어 업데이트에 대한 정보 및 장치 정보를 포함하는 블록을 생성하는지 결정하기 위하여, 업데이트된 장치 정보를 수신한 것에 응답하여, 다수의 네트워크 노드들에 의하여, 작업 증명 처리(proof-of-work processing)를 수행(perform)하는 단계를 포함할 수 있다. 다른 예에서, 소프트웨어 업데이트를 개시하는 단계는 블록체인에 가장 최근에 추가된 블록(latest added block)에 대한 이전 블록 식별자(previous block identifier)를 블록체인으로부터 획득하는 단계를 포함할 수 있고, 블록 식별자는 블록에 포함된 정보에 해시 함수를 적용하고 블록에 이전 블록 식별자를 저장하는 것에 의해 생성될 수 있다. 다수의 네트워크 노드들은 피어 노드들의 공중 네트워크(public network) 또는 피어 노드들의 사설 네트워크(private network) 중 하나에 포함될 수 있다. 블록의 수락을 결정하고 블록이 수락되는 경우 블록은 블록체인에 추가하도록 구성된 다수의 네트워크 노드들로 전송될 수 있다. 일 예에서, 블록은 인터셉트되는(intercepted) 경우, 악용(exploitation)으로부터 블록에 포함된 정보를 보호하기 위해 피어 노드의 네트워크에 블록을 전송하기 전에 암호화될 수 있다. 피어 노드들의 네트워크에 의한 블록의 수락은 블록을 수신한 과반수(majority)의 피어 노드들이 블록을 수락한다는 동의(agreement)를 통신(communicate)했다는 결정에 기초할 수 있다.

도 6은 이 기술의 모듈들이 실행될 수 있는 컴퓨팅 장치(610)를 도시한다. 기술의 높은 수준의 예가 실행될 수 있는 컴퓨팅 장치(610)가 예시되어 있다. 컴퓨팅 장치(610)는 메모리 장치들(620)과 통신하는 하나 이상의 프로세서들(612)을 포함할 수 있다. 컴퓨팅 장치(610)는 컴퓨팅 장치의 구성요소들을 위한 로컬 통신 인터페이스(618)를 포함할 수 있다. 예를 들어, 로컬 통신 인터페이스(618)는 로컬 데이터 버스 및/또는 원하는 대로 임의의 관련 주소(any related address) 또는 제어 버스들일 수 있다.

메모리 장치(620)는 프로세서(들)(612)에 의해 실행 가능한 모듈들(624) 및 모듈들(624)에 대한 데이터를 포함할 수 있다. 모듈들(624)은 앞서 설명한 기능들을 실행할 수 있다. 데이터 저장소(622)는 또한 프로세서(들)(612)에 의해 실행가능한 운영 체제와 함께 모듈들(624) 및 다른 응용프로그램들과 관련된 데이터를 저장하기 위해 메모리 장치(620)에 위치할 수 있다.

다른 응용 프로그램들도 메모리 장치(620)에 저장될 수 있고 프로세서(들)(612)에 의해 실행될 수 있다. 방법들의 하이브리드를 사용하여 컴파일, 해석 또는 실행되는 높은 수준의 프로그래밍 언어를 사용하여 소프트웨어 형태로 구현될 수 있는 이 설명에서 설명하는 구성요소들 또는 모듈들이다.

컴퓨팅 장치는 또한 컴퓨팅 장치에 의해 사용 가능한 I/O(입력/출력) 장치들(614)에 액세스할 수 있다. 네트워킹 장치들(616) 및 유사한 통신 장치들이 컴퓨팅 장치에 포함될 수 있다. 네트워킹 장치들(616)은 인터넷, LAN, WAN 또는 다른 컴퓨팅 네트워크에 연결하는 유선 또는 무선 네트워킹 장치들일 수 있다.

메모리 장치(620)에 저장된 것으로 도시된 구성요소들 또는 모듈들은 프로세서(들)(612)에 의해 실행될 수 있다. "실행 가능"이라는 용어는 프로세서(612)에 의해 실행될 수 있는 형태의 프로그램 파일을 의미할 수 있다. 예를 들어, 높은 수준의 언어의 프로그램은 메모리 장치(620)의 랜덤 액세스 부분에 로드되고(loaded) 프로세서(612)에 의해 실행될 수 있는 형식으로 기계 코드로 컴파일될 수 있거나, 소스 코드는 다른 실행 가능한 프로그램에 의해 로드되고 프로세서에 의해 실행될 메모리의 랜덤 액세스 부분에서 명령어들을 생성하도록 해석될 수 있다. 실행 가능한 프로그램은 메모리 장치(620)의 임의의 부분 또는 구성요소에 저장될 수 있다. 예를 들어, 메모리 장치(620)는 랜덤 액세스 메모리(RAM), 읽기 전용 메모리(ROM), 플래시 메모리, 솔리드 스테이트 드라이브, 메모리 카드, 하드 드라이브, 광 디스크, 플로피 디스크, 자기 테이프, 또는 임의의 다른 메모리 구성요소들일 수 있다.

프로세서(612)는 다수의 프로세서들을 나타낼 수 있고 메모리 장치(620)는 처리 회로들과 병렬로 동작하는 다수의 메모리 유닛들을 나타낼 수 있다. 이것은 시스템의 프로세스들 및 데이터에 대한 병렬 처리 채널들을 제공할 수 있다. 로컬 통신 인터페이스(618)는 임의의 다수의 프로세서들과 다수의 메모리들 간의 통신을 용이하게 하는 네트워크로서 사용될 수 있다. 로컬 통신 인터페이스(618)는 로드 밸런싱(load balancing), 벌크 데이터 전송(bulk data transfer) 및 유사한 시스템들과 같은 통신을 조정하도록 설계된 추가 시스템들을 사용할 수 있다.

이 기술에 대해 제시된 순서도들은 특정 실행 순서를 암시할 수 있지만 실행 순서는 도면과 다를 수 있다. 예를 들어, 두 개 이상의 블록들의 순서는 표시된 순서에 따라 재정렬될 수 있다. 또한, 연속적으로 도시된 두 개 이상의 블록들은 병렬로 또는 부분 병렬화로 실행될 수 있다. 일부 구성들에서는 순서도에 표시된 하나 이상의 블록들을 생략하거나 건너뛸 수 있다. 향상된 유틸리티, 계정, 성능, 측정, 문제 해결을 위해 또는 유사한 이유들로 카운터들, 상태 변수들, 경고 세마포어들 또는 메시지들을 원하는 수만큼 논리 흐름에 추가할 수 있다.

이 명세서에 설명된 기능 유닛들 중 일부는 구현 독립성을 보다 구체적으로 강조하기 위해 모듈들로 표시되었다. 예를 들어, 모듈은 맞춤형 VLSI 회로들 또는 게이트 어레이들, 논리 칩들, 트랜지스터들 또는 기타 개별 구성요소들과 같은 기성품 반도체들을 포함하는 하드웨어 회로로 구현될 수 있다. 모듈은 또한 필드 프로그램 가능 게이트 어레이들, 프로그램 가능 어레이 논리, 프로그램 가능 논리 장치들 등과 같은 프로그램 가능 하드웨어 장치들로 구현될 수 있다.

모듈들은 다양한 유형들의 프로세서들에서 실행하기 위해 소프트웨어로 구현될 수도 있다. 실행 코드의 식별된 모듈은 예를 들어 객체, 절차 또는 기능으로 구성될 수 있는 컴퓨터 명령어들의 하나 이상의 블록들을 포함할 수 있다. 그럼에도 불구하고, 식별된 모듈의 실행 파일들(executables)은, 물리적으로 함께 위치할 필요는 없지만, 논리적으로 함께 결합(join)되는 경우, 모듈을 구성하고 모듈에 대해 명시된 목적을 달성하는, 다른 위치들에 저장된 이종 명령어들(disparate instructions)을 포함할 수 있다.

실제로, 실행 가능한 코드의 모듈은 단일 명령어 또는 여러 명령어들이 될 수 있으며 여러 다른 코드 세그먼트들에 걸쳐, 여러 프로그램들 간에 그리고 여러 메모리 장치들에 걸쳐 분산될 수도 있다. 유사하게, 동작 데이터는 본 명세서에서 모듈들 내에서 식별 및 예시될 수 있으며 임의의 적절한 형태로 구현될 수 있고 임의의 적절한 유형의 데이터 구조 내에서 구성될 수 있다. 동작 데이터는 단일 데이터 세트로 수집되거나 다른 저장 장치들을 포함하는 다양한 위치들에 분산될 수 있다. 모듈들은 원하는 기능들을 수행하도록 동작시킬 수 있는 에이전트들(agents)을 포함하여 수동(passive) 또는 능동(active)일 수 있다.

본 명세서에 설명된 기술은 컴퓨터 판독 가능한 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보 저장을 위한 임의의 기술로 구현된 휘발성 및 비휘발성, 이동식 및 비이동식 매체를 포함하는 컴퓨터 판독 가능한 저장 매체에 저장될 수도 있다. 컴퓨터 판독 가능한 저장 매체는 RAM, ROM, EEPROM, 플래시 메모리 또는 기타 메모리 기술, CD-ROM, DVD(디지털 다목적 디스크) 또는 기타 광학 저장 장치, 자기 카세트, 자기 테이프, 자기 디스크 저장 장치 또는 기타와 같은 비-일시적 기계 판독 가능한 저장 매체 자기 저장 장치, 또는 원하는 정보 및 기술된 기술을 저장하는 데 사용할 수 있는 기타 컴퓨터 저장 매체를 포함하지만 이에 국한되지는 않는다.

본 명세서에 설명된 장치들은 또한 장치들이 다른 장치들과 통신할 수 있도록 하는 통신 연결들 또는 네트워킹 장치 및 네트워킹 연결들을 포함할 수 있다. 통신 연결들은 통신 매체의 일 예이다. 통신 매체는 일반적으로 컴퓨터 판독 가능한 명령어들, 데이터 구조들, 프로그램 모듈들 및 기타 데이터를 반송파(carrier wave) 또는 기타 전송 메커니즘과 같은 변조된 데이터 신호로 구현하고 임의의 정보 전달 매체를 포함한다. "변조된 데이터 신호"는 신호의 정보를 인코딩하는 방식으로 하나 이상의 특성들이 설정되거나 변경된 신호를 의미한다. 제한이 아닌 예로서, 통신 매체는 유선 네트워크 또는 직접 유선 연결과 같은 유선 매체 및 음향, 무선 주파수, 적외선 및 기타 무선 매체와 같은 무선 매체를 포함한다. 본 명세서에서 사용되는 컴퓨터 판독 가능한 매체라는 용어는 통신 매체를 포함한다.

도면들에 예시된 예들을 참조하고 동일한 내용을 설명하기 위해 본 명세서에서 특정한 언어가 사용되었다. 그럼에도 불구하고 기술의 범위에 대한 제한이 의도되지 않음을 이해해야 한다. 본 명세서에 예시된 특징들의 변경들 및 추가 수정들 및 본 명세서에 예시된 예시의 추가 적용들은 설명의 범위 내에서 고려되어야 한다.

또한, 설명된 특징들, 구조들, 또는 특성들은 하나 이상의 예들에서 임의의 적절한 방식으로 조합될 수 있다. 이전의 설명에서, 설명된 기술의 예들에 대한 철저한 이해를 제공하기 위해 다양한 구성들의 예와 같은 수많은 특정 세부사항들이 제공되었다. 그러나 이 기술은 하나 이상의 특정 세부사항들 없이 또는 다른 방법들, 구성요소들, 장치들 등을 사용하여 실행될 수 있음이 인식될 것이다. 다른 예들에서, 잘 알려진 구조들 또는 동작들은 기술의 양태들을 모호하게 하는 것을 피하기 위해 자세히 도시되지 않거나, 설명되지 않는다.

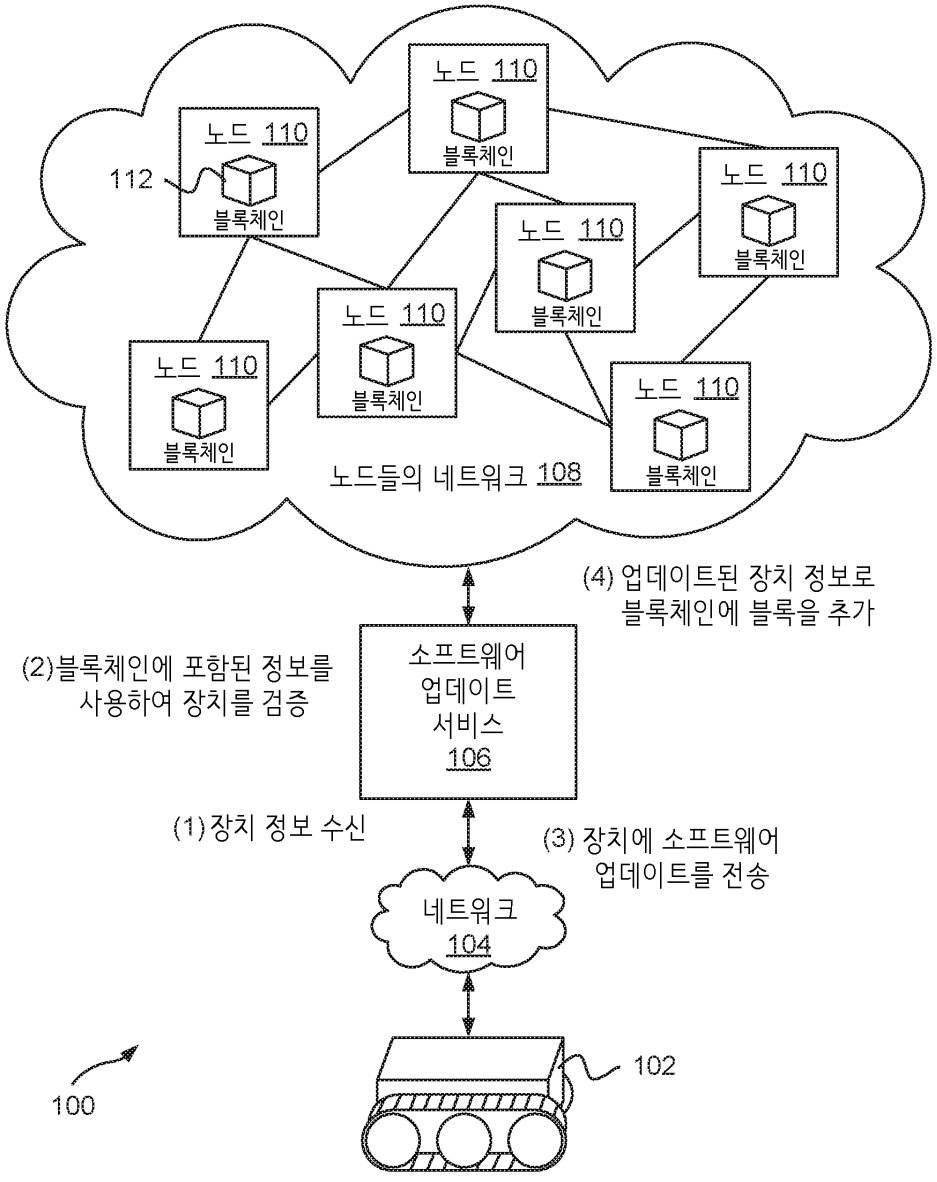
본 개시 내용은 본 명세서에 설명된 일부 예들 또는 특징들이 본 명세서에 설명된 다른 예들 또는 특징들과 결합될 수 있다는 것을 명시적으로 개시하지 않을 수 있지만, 본 개시는 당업자에 의해 실행될 수 있는 임의의 그

러한 조합을 설명하기 위해 읽혀야 한다. 본 개시에서 "또는"의 사용은 본 명세서에 달리 지시되지 않는 한 비배타적이거나, 즉 "및/또는"을 의미하는 것으로 이해되어야 한다.

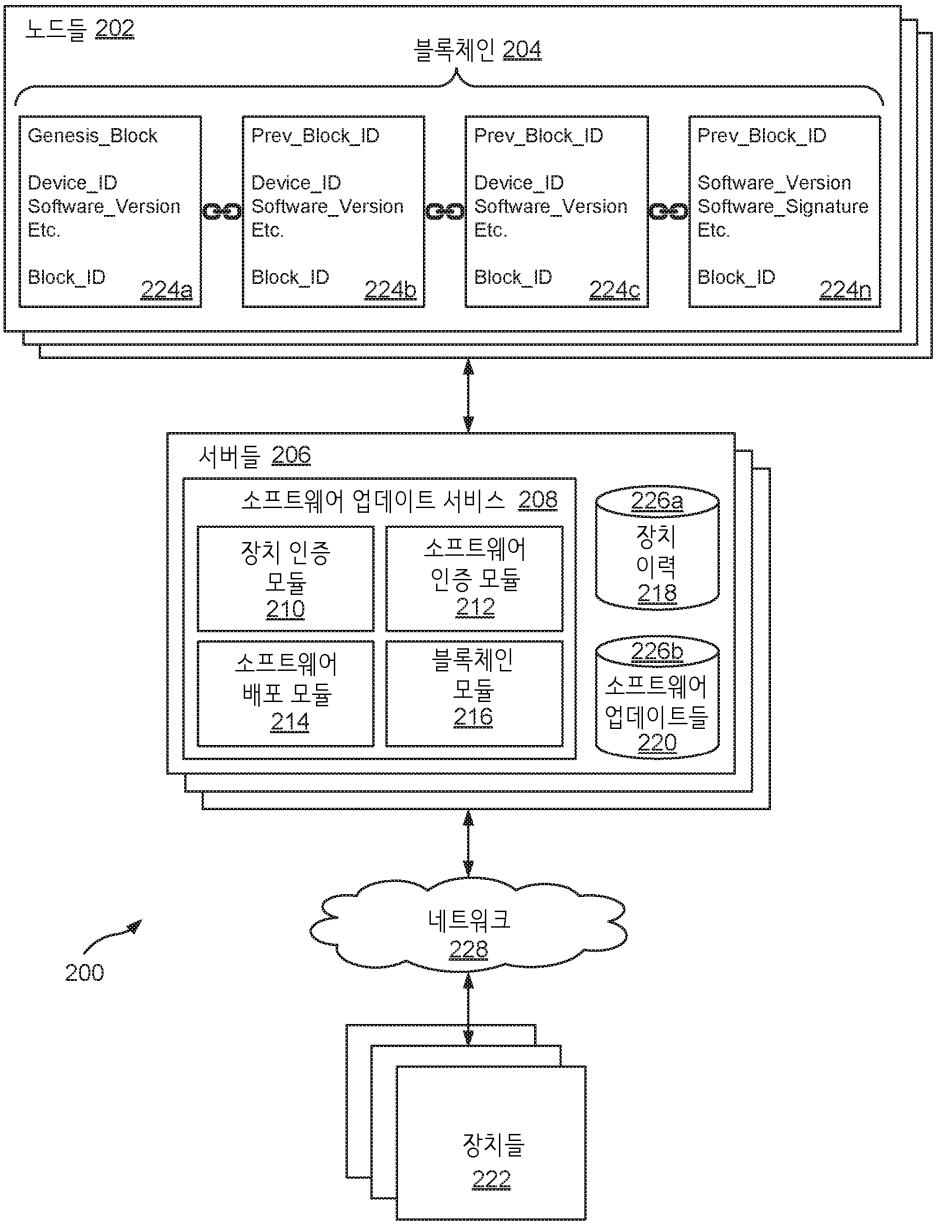
주제가 구조적 특징들 및/또는 동작들에 특정한 언어로 설명되었지만, 첨부된 청구범위에 정의된 주제가 반드시 위에서 설명된 특정한 특징들 및 동작들로 제한되지 않는다는 것을 이해해야 한다. 오히려, 위에서 설명된 특정한 특징들 및 행위들은 청구범위를 구현하는 예시적인 형태들로서 개시된다. 설명된 기술의 정신 및 범위를 벗어나지 않으면서 수많은 수정들 및 대안적인 배열들이 고안될 수 있다.

도면

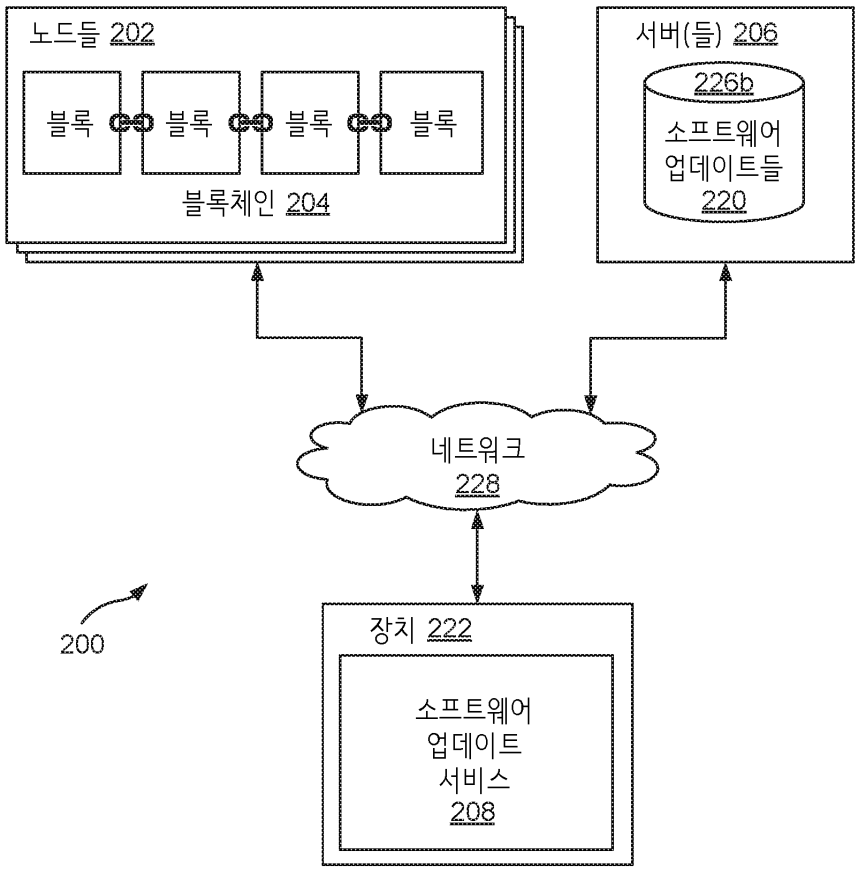
도면1



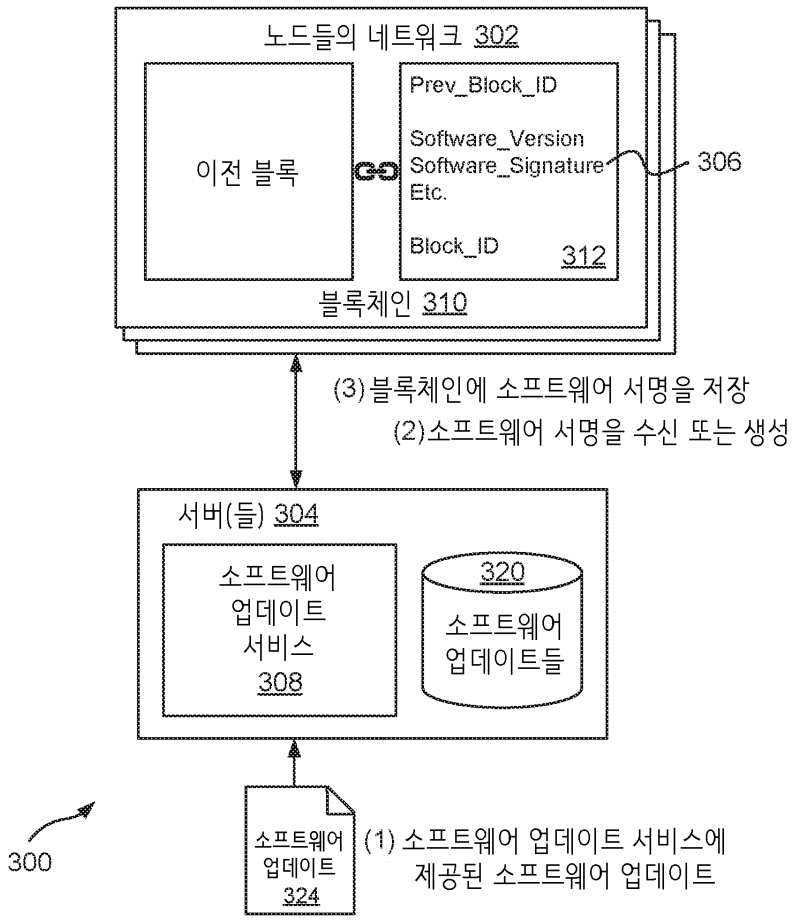
도면2a


노드들 202
블록체인 204
Genesis_Block
Device_ID
Software_Version
Etc.
Block_ID
224a
Prev_Block_ID
Device_ID
Software_Version
Etc.
Block_ID
224b
Prev_Block_ID
Device_ID
Software_Version
Etc.
Block_ID
224c
Prev_Block_ID
Software_Version
Software_Signature
Etc.
Block_ID
224n
서버들 206
소프트웨어 업데이트 서비스 208
장치 인증 모듈 210
소프트웨어 인증 모듈 212
소프트웨어 배포 모듈 214
블록체인 모듈 216
226a
장치 이력 218
226b
소프트웨어 업데이트들 220
네트워크 228
장치들 222
200

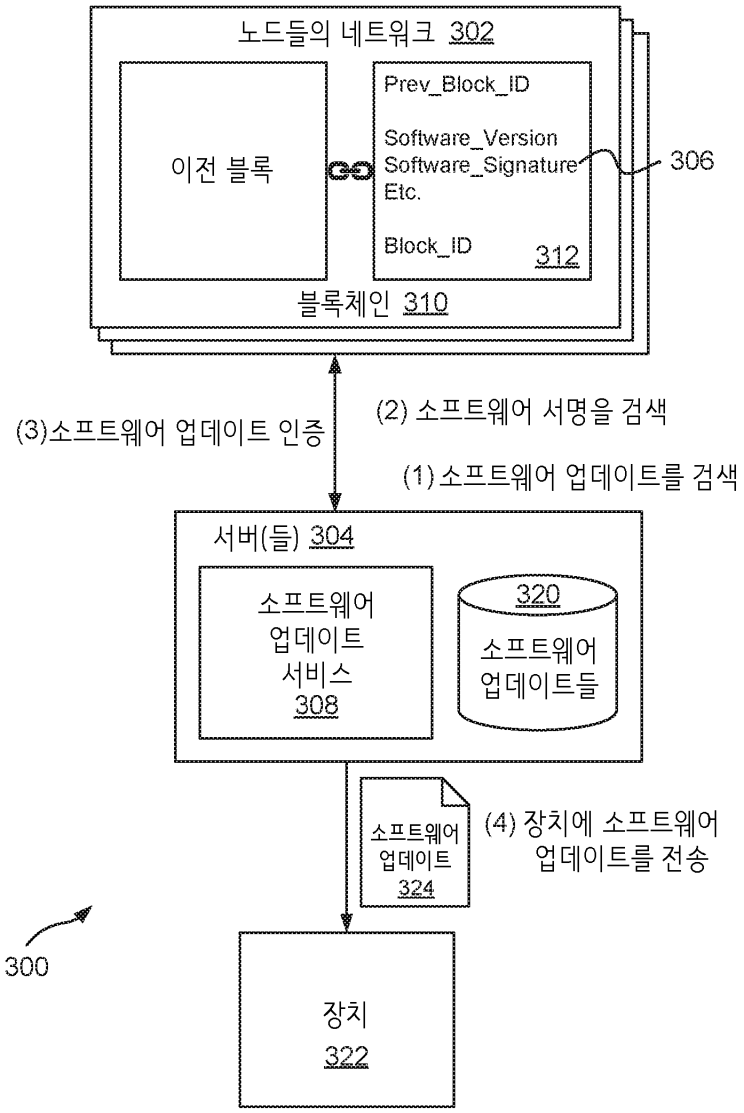
도면2b


노드들 202
블록
블록
블록
블록
블록체인 204
서버(들) 206
226b
소프트웨어
업데이트들
220
네트워크
228
200
장치 222
소프트웨어
업데이트
서비스
208

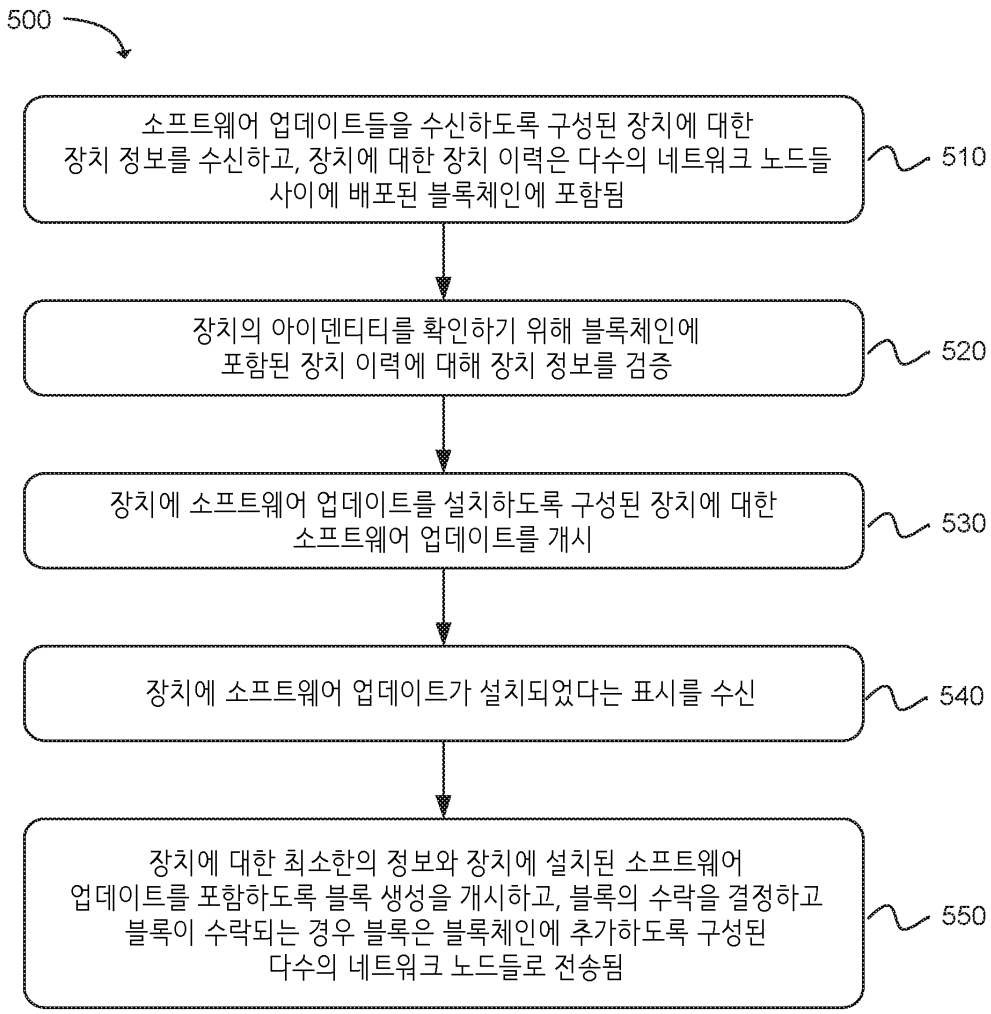
도면3


노드들의 네트워크 302
이전 블록
Prev_Block_ID
Software_Version
Software_Signature
Etc.
Block_ID
312
306
블록체인 310
(3)블록체인에 소프트웨어 서명을 저장
(2)소프트웨어 서명을 수신 또는 생성
서버(들) 304
소프트웨어 업데이트 서비스 308
320
소프트웨어 업데이트들
300
소프트웨어 업데이트 324
(1) 소프트웨어 업데이트 서비스에 제공된 소프트웨어 업데이트

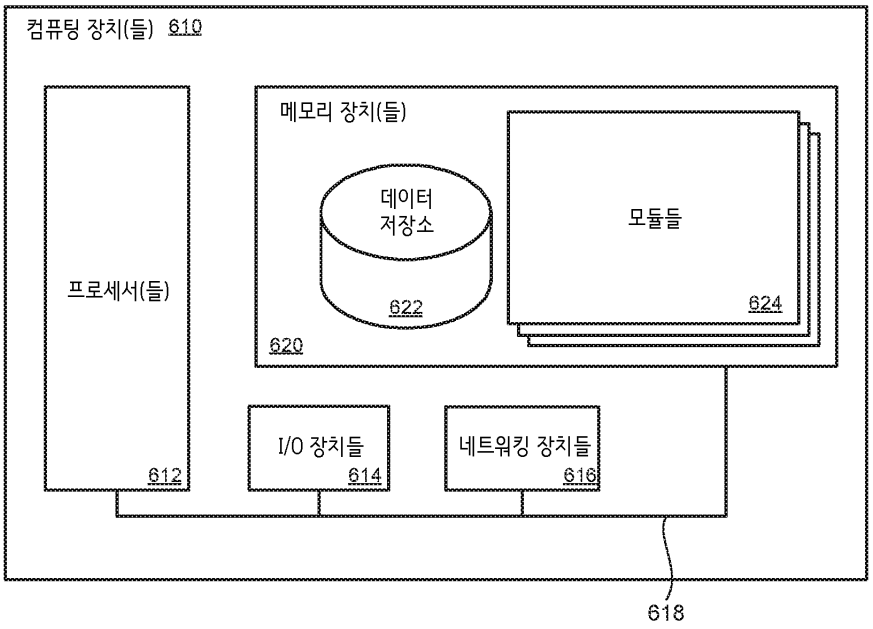
도면4


노드들의 네트워크 302
이전 블록
Prev_Block_ID
Software_Version
Software_Signature
Etc.
Block_ID
312
306
블록체인 310
(3)소프트웨어 업데이트 인증
(2) 소프트웨어 서명을 검색
(1) 소프트웨어 업데이트를 검색
서버(들) 304
소프트웨어 업데이트 서비스 308
320
소프트웨어 업데이트들
소프트웨어 업데이트 324
(4) 장치에 소프트웨어 업데이트를 전송
300
장치 322

도면5


500
소프트웨어 업데이트들을 수신하도록 구성된 장치에 대한 장치 정보를 수신하고, 장치에 대한 장치 이력은 다수의 네트워크 노드들 사이에 배포된 블록체인에 포함됨
510
장치의 아이덴티티를 확인하기 위해 블록체인에 포함된 장치 이력에 대해 장치 정보를 검증
520
장치에 소프트웨어 업데이트를 설치하도록 구성된 장치에 대한 소프트웨어 업데이트를 개시
530
장치에 소프트웨어 업데이트가 설치되었다는 표시를 수신
540
장치에 대한 최소한의 정보와 장치에 설치된 소프트웨어 업데이트를 포함하도록 블록 생성을 개시하고, 블록의 수락을 결정하고 블록이 수락되는 경우 블록은 블록체인에 추가하도록 구성된 다수의 네트워크 노드들로 전송됨
550

도면6


컴퓨팅 장치(들) 610
프로세서(들)
612
메모리 장치(들)
데이터 저장소
622
모듈들
624
620
I/O 장치들
614
네트워킹 장치들
616
618