(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0027207
(43) 공개일자 2019년03월14일

(51) 국제특허분류(Int. Cl.)
G06F 21/64 (2013.01) *G06F 21/62* (2013.01)
(52) CPC특허분류
G06F 21/64 (2013.01)
G06F 21/6209 (2013.01)
(21) 출원번호 10-2017-0113977
(22) 출원일자 2017년09월06일
심사청구일자 2017년09월06일

(71) 출원인
충남대학교산학협력단
대전광역시 유성구 대학로 99 (궁동, 충남대학교)
(72) 발명자
원유재
경기도 성남시 분당구 미금로 215, 813동 1901호 (금곡동, 청솔마을대원아파트)
박원환
서울특별시 동작구 장승배기로 26, 106동 1002호 (상도동, 아이파크상도동아파트)
백만기
대전광역시 동구 동서대로1770번길 76, 101동 503호(가양동, 한울아파트)
(74) 대리인
특허법인 공간

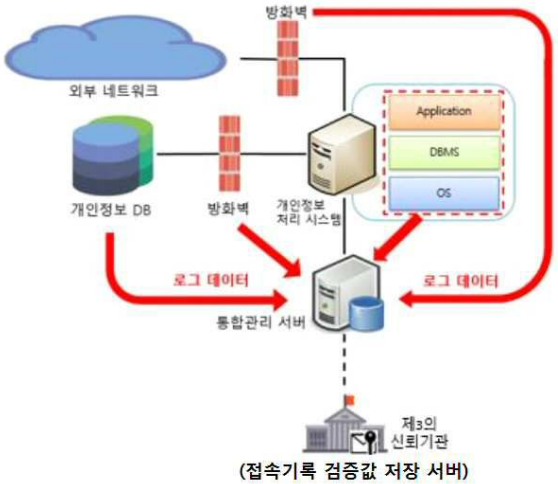
전체 청구항 수 : 총 2 항

(54) 발명의 명칭 **개인정보 접속기록 무결성 검증시스템 및 검증방법**

(57) 요 약

개인정보 접속기록 무결성 검증시스템 및 검증방법은, 초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하고, 검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생성된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하고, 무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 것을 특징으로 한다.

대 표 도 - 도5



(접속기록 검증값 저장 서버)

(52) CPC특허분류
G06F 21/6245 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 2015-0-00930
부처명 미래창조과학부
연구관리전문기관 정보통신기술진흥센터
연구사업명 SW전문인력양성사업/정보통신창의인재양성
연구과제명 SW중심대학(충남대)
기 여 율 1/1
주관기관 충남대학교 산학협력단
연구기간 2017.03.01 ~ 2017.12.31

명 세 서

청구범위

청구항 1

개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버를 포함하는 개인정보 접속기록 무결성 검증시스템으로서,

초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하고,

검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생성된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하고,

무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 것을 특징으로 하는 개인정보 접속기록 무결성 검증시스템.

청구항 2

개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버를 포함하는 개인정보 접속기록 무결성 검증시스템을 이용하는 개인정보 접속기록 무결성 검증방법으로서,

초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하는 제1 단계;

검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생선 된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하는 제2 단계;

무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 제3 단계;

를 포함하는 것을 특징으로 하는 개인정보 접속기록 무결성 검증방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 개인정보 접속기록 무결성 검증시스템 및 검증방법에 관한 것이다.

배 경 기 술

[0002] 온라인상에서 다양한 개인정보가 전송되고 있는데, 이러한 개인정보는 보호될 필요가 있고, 이러한 개인정보를 보호하기 위한 다양한 기술이 제안되고 있다.

[0003] 특허문헌 1에 개신된 기술은 그러한 기술의 예이다.

[0005] 고객의 개인정보를 통해 서비스를 제공하는 공공기관, 기업 등에서는 필수적으로 개인정보를 처리하기 위한 정보시스템(이하 ‘개인정보처리시스템’)을 운영한다. 개인정보는 개인을 알아볼 수 있거나 다른 정보와 쉽게 결합하여 개인을 알아볼 수 있는 정보이기 때문에 이 개인정보가 노출될 경우, 개인의 프라이버시를 침해할 수 있

는 위협이 존재한다. 개인정보를 통해 서비스를 제공하는 공공기관, 기업에서는 대량의 개인정보를 관리하기 때문에 침해사고가 발생하여 개인정보가 유출될 경우 수많은 사람들의 프라이버시가 침해될 수 있다. 실제로 2006년 ~ 2015년 사이에 개인정보가 대량으로 유출되는 사고가 다수 발생하였고, 이로인해 개인의 프라이버시가 위협받을 수 있는 상황이다.

[0006] 따라서 개인정보처리시스템은 개인정보가 노출되어 개인의 프라이버시가 침해되지 않도록 개인정보처리시스템 내에서 저장하고 있는 개인정보를 내외부의 공격으로부터 보호하고 사후관리를 위한 조치를 수립해야 한다. 이를 위해 NIST(National Institute of Standards Technology)에서는시스템 감사 추적(Audit Trail)을 위해 필요한 정보를 유지하여 일관된 방식으로 검토하는 것을 권고하고 있다. 또한, 우리나라에서는 개인정보보호법 등의 관련 법령에서 이와 유사하게 개인정보처리시스템에 접속하여 개인정보를 이용한 이력기록(이하 '개인정보 접속기록')을 수집하여 관리하도록 하고 있다.

[0007] 개인정보 접속기록은 개인정보의 불법 조회, 오남용 등과 같은 개인정보 침해사고의 감사 추적을 통한 침해원인 분석, 해결방안 모색 등을 목적으로 사용되며 이를 위해 필수적으로 관리해야 한다. 그러나 개인정보접속기록은 법령에서 요구하는 형식으로 생성 및 관리되는 경우도 있지만, 도 1과 같이 개인정보처리시스템을 운영하는 환경 내에 있는O/S, DBMS 등의 S/W와 방화벽과 같은 H/W 장비에서도 자체 고유의 로그기록에 법령에서 요구하는 항목(개인정보 접속기록)이 포함되어 다양한 형식으로 분산되어 생성, 관리될 수 있다.

[0008] 그 결과 법령에서 요구하는 개인정보 접속기록과 다른 성질의 로그형태로 저장된다. 그렇기 때문에 개인정보보호 현장에서 침해사고가 발생했을 경우 사고의 발생원인, 공격기법 등을 분석하여 대응하기 위해서는 도 2와 같이 별도의 통합, 추출 등의 절차를 거쳐야 하는 어려움이 있다. 따라서 각각의 로그 생성S/W 및 장치에서 생성된 로그 데이터에 포함되어 있는 개인정보 접속기록만을 별도로 추출 및 관리하여 신속하고 정확하게 침해사고를 분석해야 할 필요가 있다.

[0010] 개인정보 접속기록의 보안을 위해서, (1) MAC을 이용한 방법 (2) FssAgg 인증을 이용한 방법 (3) 전자서명을 이용한 방법 등이 제안되었다.

[0012] 개인정보 접속기록은 개인정보에 접근하여 조회, 수정 등의 행동을 기록한 것으로써 시스템 관리를 위해 기록한 로그 데이터에 포함되어 있다. 따라서 개인정보 접속기록에 대한 무결성을 입증하기 위해서는 기본적으로 로그 시스템에서 요구하는 보안사항들을 만족해야 한다. 개인정보 접속기록을 안전하게 관리하는 환경을 구축하기 위해 필요한 보안 요구사항은 다음과 같다.

[0014] ① 공격 시점 파악

[0015] 로그 시스템은 시스템 사용자의 행동, 시스템 이벤트 등을 기록으로 남기고 이를 관리하기 때문에 침해사고의 순차적인 공격 단계를 포함하고 있다. 따라서 이러한 정보를 통해 침해사고를 분석할 수 있다.

[0016] 하지만 공격자가 자신이 침입한 흔적을 없애기 위해 로그를 위변조하거나 삭제하는 등의 공격을 한다면 침해사고를 정확하고 빠르게 분석하기 힘들 뿐만 아니라 로그 분석에 대한 의미도 사라질 수 있다. 그렇기 때문에 로그를 위변조 또는 삭제하는 공격의 시점을 파악하여 공격당한 로그와 공격당하지 않은 로그를 구분할 수 있어야 한다. 이를 구분할 수 있으면 침해사고를 더욱 빠르고 정확하게 분석할 수 있다.

[0017] FssAgg 인증 기법의 경우 데이터가 생성되면 무결성 입증을 위한 통합 검증값을 생성하고, 무결성 입증 시 저장되어 있던 데이터를 통해 통합 검증값을 재생성하여 이전에 생성된 통합 검증값과 비교한다. 이로써 로그 데이터에 대한 위변조 또는 삭제공격이 발생한 것을 탐지할 수 있다. 그러나 통합 검증값은 데이터 전체를 하나의 값으로 함축시켜 검증하기 때문에 로그 데이터에 대한 공격을 탐지할 수 있지만 공격이 어느 시점에 일어났는지 알 수가 없다. 또한 관련 연구에서 MAC을 이용한 제안방법과 전자서명을 이용한 제안방법에서는 데이터가 생성되면 비밀키와 개인키를 사용하여 로그 데이터 각각의 개별 검증값을 생성하고, 검증 시에는 저장되어 있는 로그 데이터를 통해 각각의 개별 검증값을 재생성하여 이전에 생성되어 있던 검증값과 비교를 통해 무결성을 입증한다. 이를 통해 로그 데이터에 대한 위변조 또는 삭제 공격이 발생할 경우 공격이 발생한 시점을 정확하게 파악할 수 있다. 하지만 무결성 검증시 모든 데이터에 대한 개별 검증값을 생성하고 이를 이전에 생성한 검증값과 비교하는 작업을 일일히 해주어야 하기 때문에 검증시간이 매우오래 걸릴 뿐만 아니라 비효율적이다. 따라서 이

두가지의 단점을 절충할 수 있는 방안이 필요하다.

[0018] ② 전방보안과 흐름보안

[0019] 개인정보처리시스템은 다수의 개인정보를 관리하고 이를 통해 고객에게 서비스하는 역할을 한다. 만약 개인정보처리시스템이 공격당하여 개인정보가 유출된다면 개인정보를 통해 개인의 프라이버시를 침해하는 2차 피해가 발생될 수 있다. 따라서 개인정보처리시스템에 대한 보호 대책을 마련하는 것이 중요하다. 하지만, 불가피하게 개인정보처리시스템에 대한 공격이 수행되고, 침입자가 자신이 침입한 흔적을 없애기 위해 개인정보 접속기록과 더불어 개인정보 접속기록에 대한 무결성을 입증하기위한 검증값까지 조작한다면 이를 탐지하는 것이 어렵다. 따라서 공격자가 침입하였을 경우에 침입 시점 이전의 검증값을 위변조하지 못하도록 무결성 검증값에 대한 보호조치를 해야 한다. 이러한 보호조치를 '전방 보안(Forward Security)'이라고 하며, M. Bellare 등이 로그 시스템을 공격한 시점 이전에 생성된 로그 데이터의 검증값을 보호할 수 있어야 한다고 주장했다. 전방 보안은 도 3과 같이 개별 검증값을 생성할때 사용하는 비밀키 혹은 개인키(Ki)를 로그 데이터(Li)마다 다른 값으로 생성하고, 해시 함수의 일방향성을 이용하여 이전의 개별 검증값을 생성하기 위해 사용했던 키를 통해 다음 키를 도출할 수 있지만 도출된 키로부터 이전 키를 유추할 수 없도록하는 성질을 말한다.

[0020] 침입자는 공격한 시점의 키 정보를 탈취한다 하더라도 이전 검증값을 생성하기 위한 키를 알 수 없다. 또한 키 정보를 모르기 때문에 로그 데이터에 대한 검증값을 침입자의 임의대로 생성, 조작할 수 없다. 따라서 공격 시점 이전에 생성된 로그 데이터의 위변조 공격을 하지 못하도록하는 것이 가능하다.

[0021] 전방 보안을 만족했다 하더라도 로그 데이터 항목을 선택적으로 삭제하거나 로그 데이터의 순서를 바꾸는 형태의 공격이 이루어질 경우 이를 탐지하는데 어려움이 있을 수 있다. 따라서 로그 데이터의 순차적인 흐름에 대한 무결성을 입증할 필요가 있다. 이러한 보호조치를 '흐름 보안(Stream Security)'이라고 하며, A. Attila 등이 공격한 시점 이전에 생성된 로그 데이터를 보호해야 할 뿐만 아니라 로그 데이터의 순서에 대한 보호도 같이 이루어져야 한다고 주장했다. 흐름 보안은 이전에 생성된 로그 데이터와의 종속성을 부여함으로써 만족할 수 있다. FssAgg MAC 인증 기법에서는 흐름 보안을 만족하기 위하여 통합 검증값을 생성할 때 이전에 생성한 통합 검증값과 새로 생성된 로그 데이터의 개별 검증값을 결합하고, 결합한 값을 해시함수를 통해 새로운 통합 검증값을 생성함으로써 이전 로그 데이터와의 종속성을 부여하였다. 만약 데이터의 순서를 재배열하는 행위를 할 경우 로그 데이터가 생성된 시점에서 생성한 검증값과 무결성 검증을 위해 새로 생성한 검증값이 다르기 때문에 로그 데이터에 대한 재배열 공격이 발생했다는 것을 탐지할 수 있다.

[0022] ③ 로그 데이터의 위변조 및 삭제 공격 탐지

[0023] 로그 데이터의 무결성을 입증하기 위해서는 로그 데이터에 대한 위변조 또는 삭제 공격을 탐지할 수 있어야 한다. 이러한 공격을 탐지하기 위해 많은 보안 로깅 기법에서는 도 4와 같이 로그 데이터가 생성될 때마다 비밀키 또는 개인키를 이용하여 MAC 또는 서명값과 같은 개별 검증값을 생성한다. 또한 이를 통합적으로 검증할 수 있도록 해시 함수를 이용하여 통합 검증값을 생성하는 방법을 사용한다.

[0024] 개별 검증값은 각 로그 데이터(Li)에 대한 위변조 공격을 탐지할 수 있다. 각 로그 데이터가 발생하였을 때 비밀키 또는 개인키를 이용하여 개별 검증값을 생성한 후, 로그 데이터에 대한 무결성을 검증하고자 할 경우 데이터 발생 시 생성한 검증값과 저장되어 있던 로그 데이터를 통해 새로 생성한 검증값을 비교하여 일치하는지 확인함으로써 검증할 수있다. 만약 이전에 생성된 검증값과 새로 생성한 검증값이 같다면 로그 데이터에 대한 공격이 발생하지 않은 것을 의미하며, 두 값이 다르다면 로그 데이터에 대한 공격이 발생했다는 것을 의미한다. 하지만, 개별 검증값은 각 로그 데이터에 대한 무결성만 입증하기 때문에 로그 데이터와 이에 해당하는 개별 검증값의 일부를 삭제하는 경우, 이를 탐지하기가 어렵다.

[0025] 통합 검증값은 각 로그 데이터에 대한 위변조 및 삭제 공격을 탐지할 수 있다. 각 로그 데이터(Li)가 발생하였을 경우 로그 데이터들을 하나의 값으로 결합하고, 결합한 값을 해시화함으로써 통합 검증값을 생성한다. 이는 해시함수의 충돌회피성을 이용한 것으로 로그 데이터가 위변조되는 것을 탐지할 수 있다. 또한, 로그 데이터를 결합함으로써 로그 데이터가 삭제되었을 경우 결합된 값이 달라지기 때문에 해시함수의 결과값도 달라진다. 따라서 로그 데이터가 발생할 시 생성한 통합 검증값과 삭제가 이루어진 후에 생성된 통합 검증값을 비교하여 로그 데이터가 삭제되는 공격을 탐지할 수 있다.

[0026] ④ 데이터의 특성 고려

[0027] 로그 데이터는 시스템의 이벤트, 사용자의 행동 등을 기록한 데이터이기 때문에 시스템이 운영됨에 따라 로그

데이터가 지속적으로 생성된다. 따라서 로그 데이터의 양은 시간이 지남에 따라 방대하게 많아진다. 따라서 로그 데이터가 지속적으로 생성되고, 데이터의 양이 방대하다는 특성을 고려한 방안을 마련하는 것이 필요하다. 개별 검증값은 각 로그 데이터에 대한 무결성을 입증할 수 있다. 하지만 로그 데이터의 양이 방대하기 때문에 각 로그 데이터의 검증값을 모두 관리하게 된다면 이 데이터들의 무결성을 검증할 시 각각의 개별 검증값을 모두 재생성 및 비교해야하기 때문에 시스템 측면에서 발생하는 부하가 크다. 또한, 개별 검증값 생성시 개인키를 이용하여 전자서명을 생성하는 검증 방법은 비밀키를 이용하여 MAC을 생성하는 방법에 비해 연산 시간이 길지만 검증값을 생성할 경우와 검증할 경우에 사용하는 키가 다르기 때문에 보안성은 높다. 그렇기 때문에 데이터가 대량으로 발생하는 로깅 시스템의 환경을 고려한 방법을 사용하는 것이 효율적이다.

[0028] 또한, 각 로그 데이터의 목적에 따라 저장되는 속성이 존재한다. 그 중, 시간정보는 시스템에 대한 침해사고 발생 시 이를 분석하는데 중요한 정보로써 사용되기 때문에 로그 데이터에 필수적으로 포함된다. 이러한 시간정보를 활용하여 로그 데이터의 검증값을 효율적으로 관리할 수 있다면 더욱 유용하게 시스템 환경을 구축할 수 있다.

선행기술문헌

특허문헌

[0030] (특허문헌 0001) 특허문헌 1 : KR 10-2014-0038576 A (공개일: 2014.03.31.)

발명의 내용

해결하려는 과제

[0031] 본 발명이 해결하고자는 하는 과제는, 공격시점 파악, 전방보안과 흐름 보안, 위변조 및 삭제공격 탐지, 데이터의 특성 고려 측면에서 만족할 만한 특성을 갖는 개인정보 접속기록 무결성 검증시스템 및 검증방법을 제공하는 것이다.

과제의 해결 수단

[0032] 본 발명에 의한 개인정보 접속기록 무결성 검증시스템은, 개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버를 포함하는 개인정보 접속기록 무결성 검증시스템으로서, 초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하고, 검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생성된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하고, 무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 것을 특징으로 한다.

[0033] 본 발명에 의한 개인정보 접속기록 무결성 검증방법은, 개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버를 포함하는 개인정보 접속기록 무결성 검증시스템을 이용하는 개인정보 접속기록 무결성 검증방법으로서, 초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하는 제1 단계; 검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생선 된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하는 제2 단계; 무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 제3 단계;를 포함하는 것을 특징으로 한다.

발명의 효과

[0035] 본 발명에 의한 개인정보 접속기록 무결성 검증시스템 및 검증방법은 공격시점 파악, 전방보안과 흐름 보안, 위변조 및 삭제공격 탐지, 데이터의 특성 고려 측면에서 만족할 만한 특성을 가진다.

도면의 간단한 설명

[0037] 도 1은 분산적으로 로그를 생성 및 관리하는 개인정보처리시스템의 개념도

도 2는 개인정보 침해사고 분석의 순서도

도 3은 전방보안을 설명하기 위한 개념도

도 4는 개별 검증값과 통합 검증값의 생성을 설명하기 위한 개념도

도 5는 본 발명에 의한 개인정보 접속기록 무결성 검증시스템의 구성도

도 6은 개인정보 접속기록의 무결성 입증을 위한 개념도

도 7은 초기 정보교환 절차 1

도 8의 초기 정보교환 절차 2

도 9는 한 수집 주기의 검증값 생성 절차

도 10은 검증값 생성 절차

도 11은 검증값의 저장기간 만료시 절차

도 12는 무결성 검증시 절차 1

도 13은 무결성 검증시 절차 2

발명을 실시하기 위한 구체적인 내용

[0038] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 상세하게 설명하고자 한다. 그러나 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변환, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다. 본 발명을 설명함에 있어서 관련된 공지 기술에 대한 구체적인 설명이 본 발명의 요지를 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다.

[0039] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다.

[0041] 도 5는 본 발명에 의한 개인정보 접속기록 무결성 검증시스템의 구성도이다.

[0042] 본 발명에 의한 개인정보 접속기록 무결성 검증시스템은 개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버(제3의 신뢰기관)을 포함한다.접속기록 검증값 저장 서버는 제3의 신뢰기관에 위치하는데, 이하에서는 '제3의 신뢰기관'을 '접속기록 검증값 저장 서버'와 동일한 의미로 사용한다.

[0043] 개인정보 접속기록을 통합적으로 관리하는 경우 개인정보보호 현장에서 침해사고가 발생하였을 때 사고의 발생원인, 공격기법 등을 빠르고 정확하게 분석하여 대처할 수 있다. 하지만, 개인정보 접속기록을 통합적으로 관리함에 따라 개인정보 접속기록이 공격 대상이 될 수 있다. 따라서 통합된 개인정보 접속기록에 대한 보호조치가 필요하며 개인정보 접속기록은 법적인 증거자료로써 사용될 수 있기 때문에 개인정보 접속기록이 위·변조되거나 삭제당하지 않았다는 무결성을 입증하는 것이 매우 중요하다. 본 발명에서는 도 5와 같이 기존의 개인정보 접속기록이 분산적으로 저장되어 있는 환경에서 다양한 로그생성 S/W 및 장치로부터 개인정보 접속기록을 수집하고 이를 관리하기 위하여 통합관리 서버를 운영한다. 또한, 통합관리 서버에서 관리되는 개인정보 접속기록에 대한 내외부적인 공격으로부터 검증값을 보호하기 위해 제3의 신뢰기관(접속기록 검증값 저장서버)을 운영한다. 여기에서 제3의 신뢰기관은 안전하다고 가정한다.

[0044] 통합관리 서버와 제3의 신뢰기관을 통해 개인정보 접속기록의 무결성을 입증하기 위해서 도 6과 같이 로그 생성 S/W 및 장치로부터 수집 및 추출된 개인정보 접속기록을 통합관리 서버에서 관리하며, 개인정보 접속기록에 대한 검증값을 생성한 후 이를 제3의 신뢰기관으로 전송한다. 개인정보 접속기록이 발생함에 따라 생성한 검증값을 제3의 신뢰기관이 관리함으로써 통합관리 서버는 내외부적인 공격으로부터 검증값을 안전하게 보관할 수 있다. 또한 제3의 신뢰기관에서 검증값을 관리하기 때문에 공개적인 검증도 가능하다.

[0045] 통합관리 서버는 각 로그생성 S/W 및 장치로부터 개인정보 접속기록이 포함된 모든 로그 데이터를 수집한다. 그리고 로그 데이터에서 법적으로 요구하는 개인정보 접속기록을 추출하여 통합적으로 저장한다. 이를 위해 본 출원에서는 이러한 과정이 주기적으로 반복된다고 가정한다. 즉, 개인정보 접속기록이 주기적으로 통합관리 서버에 축적되는 것을 전제로 한다. 개인정보 접속기록이 통합적으로 수집될 때마다 통합관리 서버는 이를 검증하는 값을 생성한다. 이럴 경우 검증값은 한번의 수집 주기를 통합적으로 검증하기 위한 값이기 때문에 수집 주기별로 검증값을 하나씩 생성한다. 제3의 신뢰기관은 여러 기관 및 기업의 통합관리 서버를 대상으로 운영되며 통합관리 서버에서 전송한 검증값들을 관리한다.

[0046] 그리고 통합관리 서버가 무결성 검증을 위하여 제3의 신뢰기관에 검증값을 요청할 경우, 이를 통합관리 서버에게 전송한다. 검증값은 법령에서 의무화하는 개인정보 접속기록의 보관기간에 따라 최소 6개월 동안 보관한다. 위와 같은 역할을 수행하는 통합관리 서버와 제3의 신뢰기관을 통해 개인정보 접속기록의 무결성을 입증하는 과정을 아래와 같이 3개의 단계로 나누어 설명할 수 있다.

[0047] ① 초기 정보교환

[0048] ② 검증값 생성

[0049] ③ 무결성 검증

[0051] (1) 초기 정보교환

[0052] 초기 정보교환 단계에서는 도 7과 같이 통합관리 서버와 제3의 신뢰기관간에 안전한 정보전달을 위해 서로의 공개키를 공유하고, 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송한다. 안전한 키 교환을 위하여 통합관리 서버와 제3의 신뢰기관은 서로간의 인증을 수행한 상태라고 가정한다.

[0053] 통합관리 서버와 제3의 신뢰기관은 서로간에 전송하는 데이터를 암호화 하기 위하여 도 8과 같이 각각 공개키(P_B, P_C)와 개인키(S_B, S_C)쌍을 생성한다. 그 후, 통합관리 서버는 제3의 신뢰기관에게 공개키를 요청하는 메시지를 전송한다. 제3의 신뢰기관은 통합관리 서버로부터 전송받은 공개키 요청 메시지를 확인하고 자신의 공개키(P_C)를 통합관리 서버에게 전송한다. 통합관리 서버는 제3의 신뢰기관으로부터 전송받은 공개키를 확인하고 자신의 공개키(P_B)를 전송한다. 이와 같은 과정을 통하여 통합관리 서버와 제3의 신뢰기관은 서로간에 전송하는 데이터를 교환한 공개키로 암호화하여 전송할 수 있게 된다. 통합관리 서버의 공개키(P_B)를 전송받은 것을 확인한 제3의 신뢰기관은 개인정보 접속기록의 개별 검증값인 MAC을 생성할 때 필요한 초기 비밀키(K_1)와 통합관리 서버를 식별할 수 있는 식별자인 ID를 통합관리 서버에게 요청하는 메시지를 전송한다. 이 메시지를 수신한 통합관리 서버는 첫번째 개별 검증값을 생성하기 위해 필요한 비밀키를 난수로 생성하고, 자신의 식별자와함께 제3의 신뢰기관의 공개키로 암호화한다. 그 다음으로 이 값을 통합관리 서버의 개인키로 전자서명하여 제3의 신뢰기관으로 전송한다. 전자서명은 서명자만이 서명값을 생성할 수 있기 때문에 위조가 불가하며, 이 값을 전송받은 수신자는 서명자의 공개키를 이용하여 서명값을 확인할 수 있기 때문에 서명자를 인증할 수 있다. 또한 전송하는 메시지가 전송간에 변경되지 않았다는 것을 증명할 수 있으며 부인방지 기능도 포함되기 때문에 더욱 안전하게 메시지를 전달할 수 있는 장점이 있다.

[0055] (2) 검증값 생성

[0056] 통합관리 서버는 로그생성 S/W 및 장치로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 로그 데이터에 포함되어 있는 개인정보 접속기록을 추출하여 통합적으로 관리한다. 이 때 로그 데이터의 수집은 주기적으로 일어나며, 검증값 또한 로그 데이터의 수집 주기별로 생성된다.

[0057] 도 9와 같이 로그생성 S/W 및 장치는 자신이 생성한 모든 로그 데이터를 일정한 주기마다 통합관리 서버에게 전

송한다. 통합관리 서버는 전송받은 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출하여 이를 통합적으로 관리할 수 있도록 하나의 저장소에 저장한다. 이렇게 개인정보 접속기록이 추출 및 통합되면 이 데이터에 대한 검증값을 생성하게 된다. 검증값은 FssAgg MAC 인증 기법과 같이 각 개인정보 접속기록에 대한 개별 검증값인 MAC 생성한다. 개별 검증값을 생성하는 방법으로 전자서명을 사용할 수 있지만, 전자서명을 통해 검증 값을 생성할 경우 MAC에 비해 처리속도가 느리기 때문에 방대한 데이터가 발생하는 환경에서는 전자서명이 비효율적이다. 따라서 MAC을 이용하여 개별 검증값을 생성한다. MAC을 생성할 때 사용하는 비밀키는 초기 정보교환 단계에서 생성한 비밀키를 해시체인 기법으로 업데이트하면서 각 개인정보 접속기록마다 다른 비밀키를 사용한다. 또한, 개별 검증값을 생성하기 위하여 사용한 비밀키는 다음 개별 검증값을 생성하기 위한 비밀키로 업데이트하고 이전 비밀키의 노출을 방지하기 위하여 삭제한다. 이럴 경우, 검증값에 대한 전방보안을 만족하며 통합관리서버에 대한 공격이 발생하더라도 공격 시점 이전의 비밀키를 유추할 수 없다. 이렇게 전방보안을 만족하는 해시체인을 적용한 비밀키와 MAC을 이용하여 개별 검증값을 도출하는 방법은 다음과 같다.

[0058] K_i : i 번째 비밀키, K_1: 난수(Seed), h(x) : 단방향 해시함수,

[0059] R_i : i 번째 개인정보 접속기록, $MAC_{ki}(x)$: MAC 함수

[0060] Z_i : i 번째 개별 검증값

[0062] $K_i = h^{i-1}(SEED)$, $Zi = MAC_{ki}(Ri)$

[0063] 개별 검증값을 생성한 후, 한 수집 주기에 생성된 개별 검증값들을 하나의 값으로 검증할 수 있도록 통합하여 검증값을 생성한다. 통합 검증값은 이전까지 생성된 통합 검증값과 개별 검증값을 해시화하여 얻은 결과값으로 새 통합 검증값을 생성하면 이전에 생성된 통합 검증값은 삭제한다. 이렇게 생성된 통합 검증값은 한 수집 주기 동안의 개인정보 접속기록에 대한 위·변조와 삭제 공격을 탐지할 수 있게 된다. 또한, 수집 주기별 생성된 개인정보 접속기록의 무결성을 입증할 수 있다. 즉, 수집 주기별로 무결성을 입증할 수 있기 때문에 어떤 수집 주기에 공격이 발생했는지 파악할 수 있게 된다. 그 후, 통합 검증값을 생성한 통합관리 서버는 통합 검증값과 자신을 식별할 수 있는 ID, 그리고 TAG라는 값을 초기 정보교환 단계에서 전송받은 제3의 신뢰기관의 공개키로 암호화한다. 그리고 ID와 TAG를 자신의 개인키로 전자서명하여 암호화를 수행한 값과 함께 제3의 신뢰기관으로 전송한다. 시간이 지남에 따라 통합 검증값은 수집 주기 별로 생성되기 때문에 관리해야할 통합 검증값의 수가 점점 증가하게 된다.

[0064] 이럴 경우, 통합관리 서버에 침입한 공격자가 수집 주기 단위로 구성된 개인정보 접속기록의 순서를 재배열 하는 공격을 한다면 무결성 검증이 제대로 이루어질 수 없다. 그렇기 때문에 통합 검증값들의 순서에 대한 무결성인 흐름 보안을 만족해야 한다. 따라서 도 10과 같이 개인정보 접속기록의 수집 주기에 따라 생성된 통합 검증값의 흐름 보안을 만족하기 위하여 TAG를 사용한다.

[0065] TAG는 이전 주기에 생성된 TAG와 현재 주기에 생성된 통합 검증값을 해시화한 결과값이다. 이렇게 값을 생성할 경우, 통합 검증값을 체인형태로 구성하여 무결성을 검증할 시 개인정보 접속기록의 순서가 재배열 되어 있다면 TAG값이 달라지기 때문에 재배열 공격을 탐지할 수 있다. 또한, TAG는 통합 검증값에 대한 무결성을 입증할 수 있다. 따라서 하나의 수집 주기에 해당하는 개인정보 접속기록을 삭제하는 공격을 탐지할 수 있다. 개별 검증값을 통해 통합 검증값과 TAG를 도출하고 이를 암호화, 전자서명하는 방법은 다음과 같다.

[0066] B : 통합관리 서버, C : 제3의 신뢰기관

[0067] Z_i : i 번째 개별 검증값, H_i : i 번째 통합 검증값,

[0068] h(x) : 단방향 해시함수, P_x : x의 공개키, S_x : x의 개인키,

[0069] $Sign_{Sx}()$: x의 개인키로 전자서명, $E_{Px}()$: x의 공개키로 암호화

[0070] TAG : 통합 검증값의 흐름보안을 위한 값, ID : 통합관리 서버 식별자

[0071] $H_i = h(Z_i \parallel H_{i-1})$

[0072] TAG_i = h(이전 수집 주기의 TAG ‖ 현재 수집 주기의 통합 검증값),

[0073] ID 및 검증값의 암호화 = Epc(ID ‖ Hi ‖ TAG)

[0074] ID 및 검증값의 전자서명 = SignSB(ID ‖ Hi ‖ TAG)

[0076] 제3의 신뢰기관은 통합관리 서버로부터 전송받은 값을 수신하여 통합관리 서버의 공개키로 서명값을 확인하고 자신의 개인키로 복호화한다. 그 다음 통합관리 서버에서 전송한 값이 올바른 값인지 확인하기 위하여 저장하고 있던 이전 수집 주기의 TAG와 현재 수집 주기에 전송받은 통합 검증값을 통해 새로운 TAG를 생성하고 현재 주기에 전송받은 TAG와 비교하여 올바른 값이 맞다면 통합 검증값과 TAG를 저장하고 이전 수집주기의 TAG는 폐기한다. 이렇게 한 수집 주기별로 생성된 통합 검증값은 제3의 신뢰기관에서 법적 의무사항인 최소 6개월이상 관리하게 된다. 또한 제3의 신뢰기관은 모든 개인정보 접속기록의 개별 검증값을 관리할 필요없이 통합검증값을 관리하기 때문에 저장소를 적게 사용하고 통신에 대한 부하도 줄어들게 된다.

[0077] 제3의 신뢰기관에서는 법령에서 요구하는 의무사항을 만족하고 저장소의 한계를 고려하여 통합관리 서버로부터 전송받은 통합 검증값 및 TAG, 그리고 초기 비밀키를 6개월 단위로 1년 동안 저장한다. 즉, 도 11과 같이 초기 비밀키가 생성된 시점부터 6개월 동안 생성된 통합 검증값과 TAG를 하나의 집합 A라고 하고 그 다음 6개월 동안 생성된 초기 비밀키와 통합 검증값, 그리고 TAG의 집합을 B라고 가정할 때, 집합 B의 생성기간이 6개월이 지나 만료되었다면 기존에 저장하고 있던 A집합을 삭제하고, B는 유지한다.

[0078] 6개월이 지나 보관기간 만료가 발생할 시 통합관리 서버는 제3의 신뢰기관에 자신의 식별자와 함께 만료 요청 메시지를 전송한다. 그 후, 집합C의 초기 비밀키($K_{a'}$)를 난수로 생성하여 암호화 및 전자서명을 적용하여 전송한다. 제3의 신뢰기관에서는 이 메시지에 대한 서명 확인 및 복호화를 통하여 메시지를 확인하고 통합관리 서버의 식별자를 확인하여 해당 통합관리 서버의 A집합을 삭제한다. 그리고 전송받은 새로운 초기 비밀키를 포함한 C집합을 생성하여 이후 통합관리 서버로부터 전송받는 통합 검증값과 TAG를 저장하게 된다. 이와 같이 6개월 단위로 초기 비밀키, 통합 검증값, TAG를 1년동안 보관하고, 6개월 단위마다 생성되는초기 비밀키는 항상 난수를 통해 생성한다. 즉, 각 집합에 포함되어 있는 초기 비밀키는 서로 연관성이 없는 값이다.

[0079] (3) 무결성 검증

[0080] 무결성 검증 단계는 통합관리 서버에 저장되어 있는 개인정보 접속기록이 변경되지 않았다는 것을 입증하는 단계이다. 즉, 개인정보 접속기록이 위·변조 또는 삭제 공격을 당하였는지 판단한다. 이 단계에서는 통합관리 서버의 필요시 검증을 수행해야 하므로 검증을 수행하는 주체는 통합관리 서버이고, 도12와 도 13과 같이 무결성 검증을 수행해야 할 시 통합관리 서버는 제3의 신뢰기관으로 자신을 식별할 수 있는 ID와 검증값을 요청하는 메시지를 전송한다.

[0081] 제3의 신뢰기관은 이 메시지를 전송 받아 통합관리 서버의 식별자인 ID를 확인하고 해당하는 통합관리 서버의 검증값(H_1 ~ H_x) 및 초기 비밀키(K_1), 그리고 TAG를 통합관리 서버의 공개키로 암호화한다. 또한 검증값(H_1 ~ H_x), 초기 비밀키(K_1), TAG를 자신의 개인키로 전자서명하여 암호화를 수행한 결과값과 함께 통합관리 서버로 전송한다. 통합관리 서버는 전송받은 메시지의 전자서명을 제3의 신뢰기관의 공개키로 전자서명을 확인하고, 암호화된 메시지는 자신의 개인키로 복호화하여 임시저장한다. 그리고 자신이 저장하고 있던 개인정보 접속기록의 통합 검증값 및 TAG를 검증값 생성 단계의 절차대로 재생성한다($H'_1,\cdots,H'_x$, TAG'_x). 그 후, 제3의 신뢰기관으로부터 전송받은 검증값들과 비교하여 일치하는지 여부를 파악하여 무결성을 입증한다. 만약 재생성한 검증값이 제3의 신뢰기관으로부터 받은 검증값과 일치하지 않는다면 개인정보 접속기록이 위·변조 또는 삭제 공격을 당했다는 것을 의미하고, 일치한다면 개인정보 접속기록에 대한 공격이 발생하지 않았다는 것을 의미한다.

[0083] 본 발명에 의한 개인정보 접속기록 무결성 검증시스템은, 개인정보 처리시스템, 통합관리 서버, 접속기록 검증값 저장 서버를 포함한다.

[0084] 초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송한다.

[0085] 검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생선 된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송한다.

[0086] 무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단한다.

[0087] 본 발명에 의한 개인정보 접속기록 무결성 검증방법은 다음의 3단계를 포함한다.

[0088] (1) 초기 정보교환을 위하여, 통합관리 서버와 접속기록 검증값 저장 서버는 서로의 공개키를 공유하고 검증값을 도출하기 위해 필요한 비밀키를 생성하여 전송하는 제1 단계

[0089] (2) 검증값 생성을 위해, 통합관리 서버는 개인정보처리시스템으로부터 생성된 모든 로그 데이터를 주기적으로 수집하고, 수집된 로그 데이터 중에서 개인정보 접속기록에 해당하는 로그 데이터만을 추출한 후, 개별 검증값 및 한 수집 주기에 생선 된 개별 검증값들을 하나의 값으로 검증할 수 있는 통합 검증값을 생성하고, 생성된 개별 검증값 및 통합 검증값을 접속기록 검증값 저장 서버로 전송하는 제2 단계

[0090] (3) 무결성 검증을 위해, 통합관리 서버는 스스로 개별 검증값 및 통합 검증값을 생성한 후, 접속기록 검증값 저장 서버로부터 전송받은 검증값과 비교하여, 일치하면 무결성이 있는 것으로 판단하는 제3 단계

[0092] 본 발명에 의한 개인정보 접속기록 무결성 검증시스템 및 검증방법이 갖는 특성을 평가하면 다음과 같다.

[0093] ① 공격시점 파악

[0094] 본 출원에서는 개인정보 접속기록의 무결성을 입증하기 위해 FssAgg MAC 인증 기법을 응용하여 무결성 검증값을 생성한다. 이 때, 개인정보접속기록의 각 수집주기마다 하나의 검증값이 생성되며 이 값을 통해 각수집주기에 생성된 개인정보 접속기록의 무결성을 입증한다. 따라서 이 검증값을 통해 어떤 수집주기에 개인정보 접속기록의 위·변조 또는 삭제공격이 발생하였는지 탐지할 수 있다. 즉, 각 수집주기에 생성된 검증값을 무결성 검증 단계에서 새로 생성된 검증값과 비교함으로써 공격 시점을 파악할 수 있다.

[0095] ② 전방보안과 흐름보안

[0096] 전방보안(Forward Security)은 시스템이 공격당한 후 검증값을 생성할 때사용되는 비밀키가 노출되었다 하더라도 공격시점 이전의 비밀키를 유추할 수 없도록 하는 성질이다. 이전에 사용된 비밀키를 유추할 수 없기때문에 공격시점 이전에 생성된 검증값의 위·변조를 할 수 없다. 따라서 특정 시점 이전의 검증값을 보호할 수 있다. 이러한 요구사항을 만족하기 위하여 본 출원에서는 해시체인 매커니즘을 통한 Key Evolution을 수행함으로써 각각의 개인정보 접속기록에 대한 MAC을 생성할 시 서로 다른 비밀키를 사용하고, 특정 시점에 사용된 비밀키를 통해 다음 비밀키를 도출할 수 있지만 다음 비밀키를 통해 이전 비밀키를 유추할 수 없도록 하여 전방보안을 만족하였다.

[0097] 흐름보안(Stream Security)은 데이터의 순서에 대한 무결성을 보호하는 방법으로써 데이터의 일부가 삭제되거나 데이터의 순서를 바꾸는 공격을 탐지할 수 있다. 이러한 흐름보안은 수집된 데이터 간 종속성을 부여함으로써 만족할 수 있다. 본 발명에서는 한 수집주기에 수집된 개인정보 접속기록을 FssAgg MAC 인증 기법으로 검증값을 생성한다. FssAgg MAC 인증 기법으로 검증값을 생성할 경우 생성된 데이터의 개별 검증값을 생성할 때마다 개별 검증값과 이전에 생성된 통합 검증값을 통해 새로운 통합 검증값을 계산하기 때문에 이전 데이터가 삭제되거나 순서가 변경될 시 이를 탐지할 수 있게 된다. 그리고 수집주기별 생성된 통합 검증값을 통해 TAG를 생성하기 때문에 하나의 수집주기에 생성된 개인정보 접속기록을 모두 삭제하거나 통합 검증값의 순서를 재배열하는 공격을 탐지할 수 있다.

[0098] ③ 위·변조 및 삭제 공격 탐지

[0099] 개인정보 접속기록을 관리해야 하는 시스템은 개인정보 접속기록이 침해사고 분석에 사용되기 때문에 데이터에 대한 보호대책이 필요하다. 따라서 개인정보 접속기록을 관리하는 시스템에서 개인정보 접속기록이 위· 변조되거나 일부 데이터가 삭제당하는 공격을 탐지할 수 있어야 한다. 이를 위해 각 수집 주기마다 무결성을 검증할 수 있는 값을 생성하여 제3 의 신뢰기관으로 전송하여 외·내부의 공격으로부터 보호한다. 그리고 무결성 검증 시에는 저장되어 있는 개인정보 접속기록을 통해 새로운 검증값을 생성하고 제3의 신뢰기관으로부터 전송받은

검증값과 비교하여 일치하는지 여부를 확인한다. 만약 두 값이 같으면 개인정보 접속기록에 대한 공격이 이루어지지 않은 것이고, 두 값이 다르다면 개인정보 접속기록에 대한 변화가 생긴 것이므로 공격을 받은 것이라 판단할 수 있다.

[0100] ④ 데이터의 특성 고려

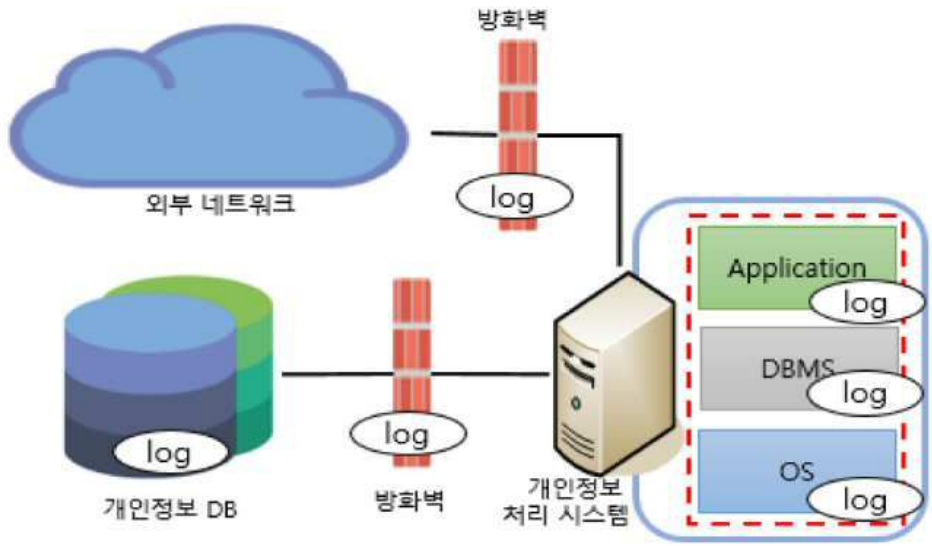
[0101] 로그 데이터는 시스템의 이벤트, 사용자의 행위 등을 기록한 정보이기 때문에 지속적으로 데이터가 발생하며 양이 매우 방대하다. 개인정보 접속기록은 로그 데이터에 포함되어 있는 정보이므로 로그 데이터의 특성을 그대로 가지고 있다. 따라서 방대한 양으로 발생하는 개인정보 접속기록의 무결성을 효율적으로 검증할 수 있어야한다. 이를 위해 대량으로 발생하는 개인정보 접속기록의 무결성을 검증할 수 있는 값을 수집주기별 하나의 값으로 생성한다. 그리고 무 결성 검증 시 각 개인정보 접속기록의 무결성을 일일히 비교하지 않고 주기별로 생성된 검증값만을 비교함으로써 주기별로 개인정보 접속기록의 무결성을 검증한다. 따라서 시스템 측면에서의 부하를 줄일 수 있다.

[0102] 기존의 제안된 보안 로깅 기법 연구중 전자서명을 이용한 제안방법을 제외하고 전방보안을 만족하기 위하여 본 연구에서 사용한 방법과 같이 Key Evolution을 해시체인 매커니즘을 사용한다. 또한, FssAgg 인증 기법을 사용한 방법에서는 모든 로그 데이터의 무결성을 하나의 검증값으로 검증하기 때문에 데이터에 대한 위·변조 또는 삭제 공격이 발생한 것은 탐지할 수 있지만 공격시점을 파악할 수 없다. 하지만, MAC을 이용한 제안방법과 전자서명을 이용한 제안방법에서는 각각의 로그 데이터에 해당하는 검증값을 생성하여 무결성 검증 시 일일히 이 값들을 비교함으로써 검증하기 때문에 로그 데이터가 공격당한 시점을 정확히 파악할 수 있다. MAC을 이용한 제안방법과 FssAgg 인증을 이용한 제안방법에서는 검증값을 생성할 시 데이터간에 종속성을 부여함으로써 데이터 흐름에 대한 무결성을 보장할 수 있는 방안을 사용하였다. 하지만 전자서명을 이용한 제안방법에서는 이러한 과정이 없기 때문에 흐름보안은 만족하지 못한다.

[0103] 또한, 기존의 제안된 방법 모두 로그 데이터의 보호를 위한 방안을 제시한 연구로써 로그 데이터의 위·변조 및 삭제 공격에 대한 탐지가 가능하다. 하지만, MAC을 이용한 제안방법에서는 로그 데이터 마지막 부분의 일부 집합을 삭제하는 공격은 탐지하지 못하기 때문에 삭제 공격에 대해서 완벽하게 탐지한다고 할 수 없다. MAC을 이용한 제안방법과 전자서명을 이용한 제안방법에서는 로그 데이터 각각에 대한 검증값을 생성하고 무결성 검증 시 새로 생성한 검증값과 기존에 생성한 검증값을 일일히 비교하기 때문에 무결성 검증 단계에서 시스템의 부하가 매우 크다. 따라서 대량으로 로그 데이터가 발생하는 환경에서의 사용이 부적합하다. 반면, FssAgg 인증을 사용한 제안방법에서는 모든 로그 데이터의 무결성검증을 하나의 통합된 값을 통해 수행하기 때문에 무결성 검증시 시스템의 부하가 발생하지 않는다.

도면

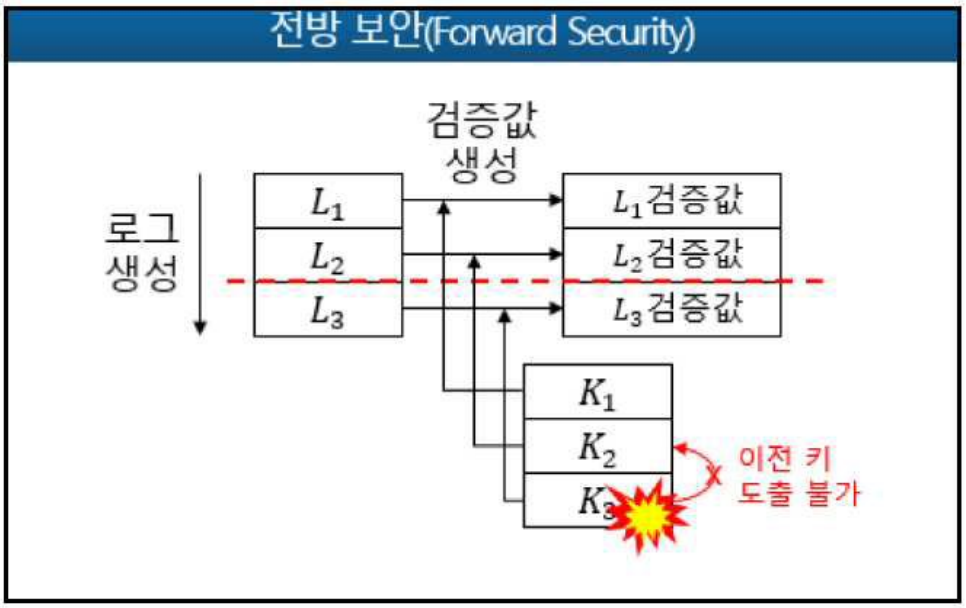
도면1



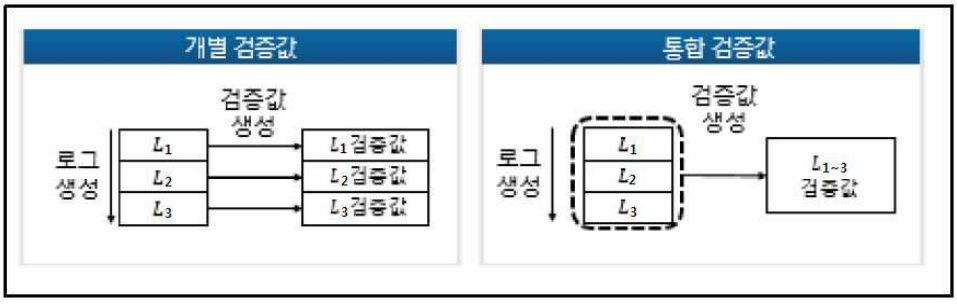
도면2


빠른 분석을
방해하는 요인
침해사고
발생
개인정보
접속기록 추출
개인정보
접속기록 통합
침해사고
분석

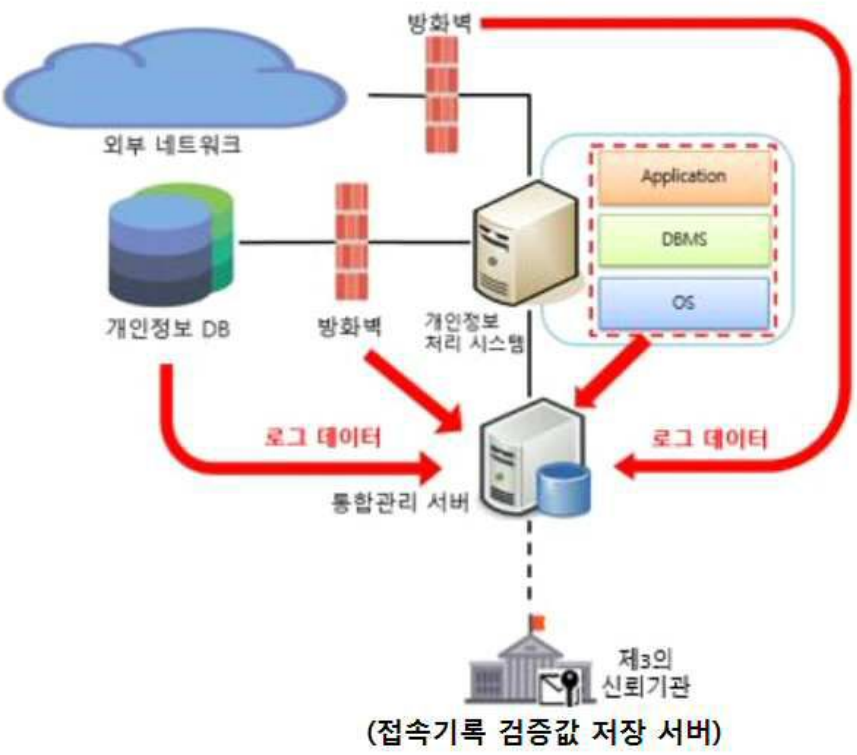
도면3


전방 보안(Forward Security)
검증값
생성
로그
생성
L_1
L_2
L_3
L_1 검증값
L_2 검증값
L_3 검증값
K_1
K_2
K_3
이전 키
도출 불가

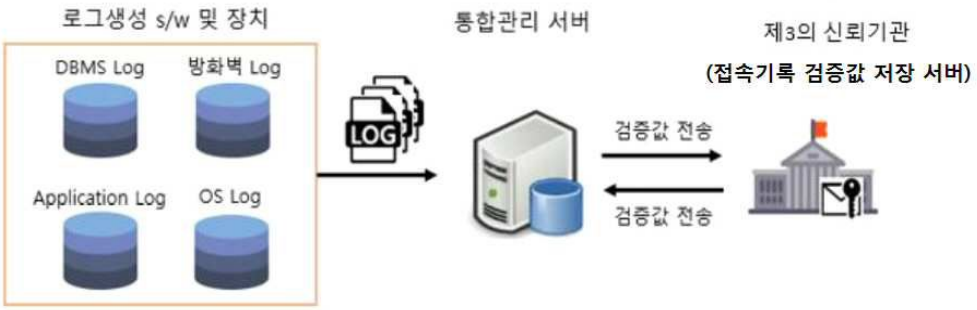
도면4


개별 검증값
검증값
생성
로그
생성
L_1
L_2
L_3
L_1검증값
L_2검증값
L_3검증값
통합 검증값
검증값
생성
로그
생성
L_1
L_2
L_3
$L_{1\sim3}$
검증값

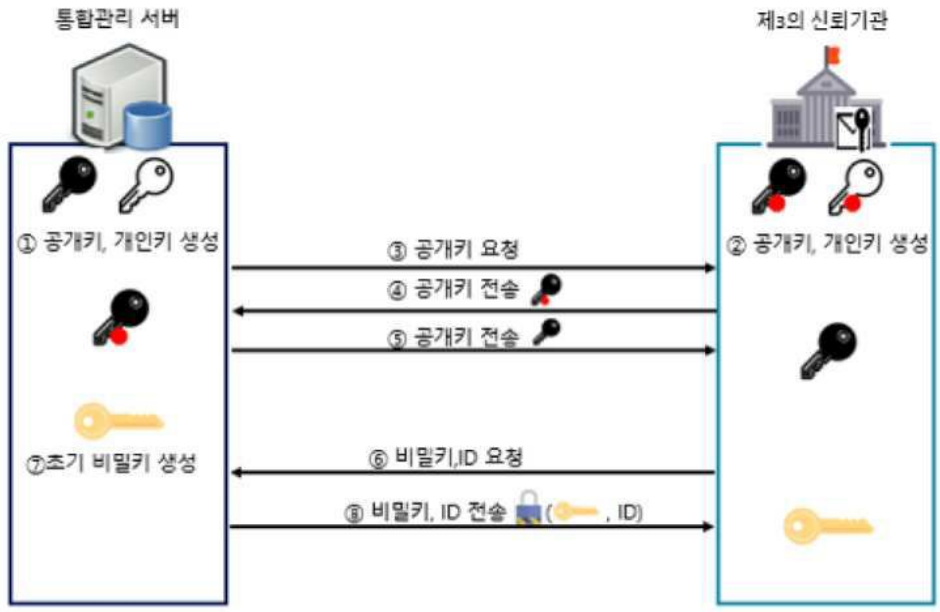
도면5


방화벽
외부 네트워크
Application
DBMS
OS
개인정보 DB
방화벽
개인정보 처리 시스템
로그 데이터
로그 데이터
통합관리 서버
제3의 신뢰기관
(접속기록 검증값 저장 서버)

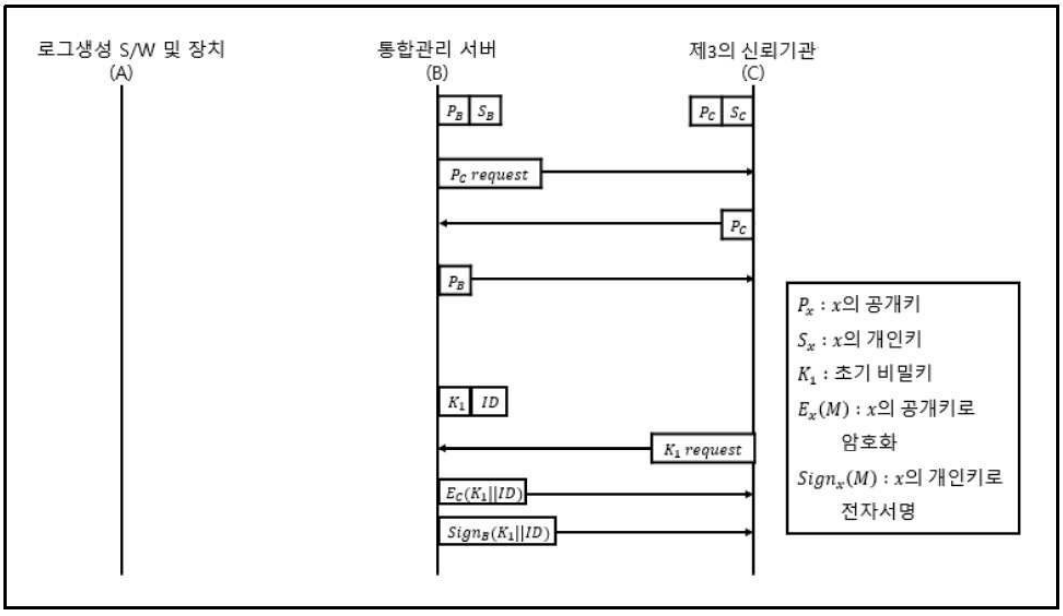
도면6


로그생성 s/w 및 장치
DBMS Log
방화벽 Log
Application Log
OS Log
LOG
통합관리 서버
검증값 전송
검증값 전송
제3의 신뢰기관
(접속기록 검증값 저장 서버)

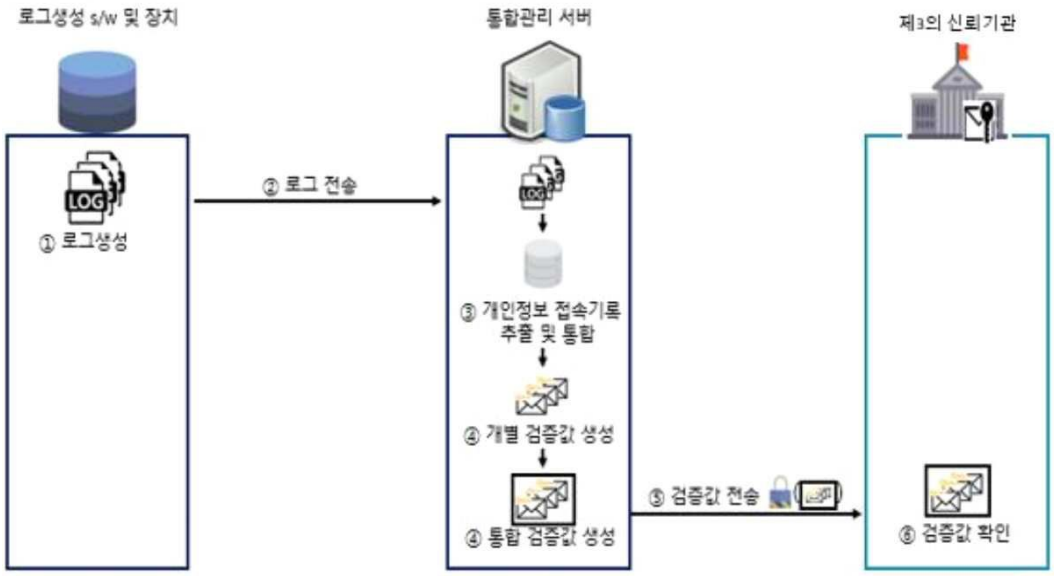
도면7


통합관리 서버
제3의 신뢰기관
① 공개키, 개인키 생성
② 공개키, 개인키 생성
③ 공개키 요청
④ 공개키 전송
⑤ 공개키 전송
⑥ 비밀키,ID 요청
⑦초기 비밀키 생성
⑧ 비밀키, ID 전송 (, ID)

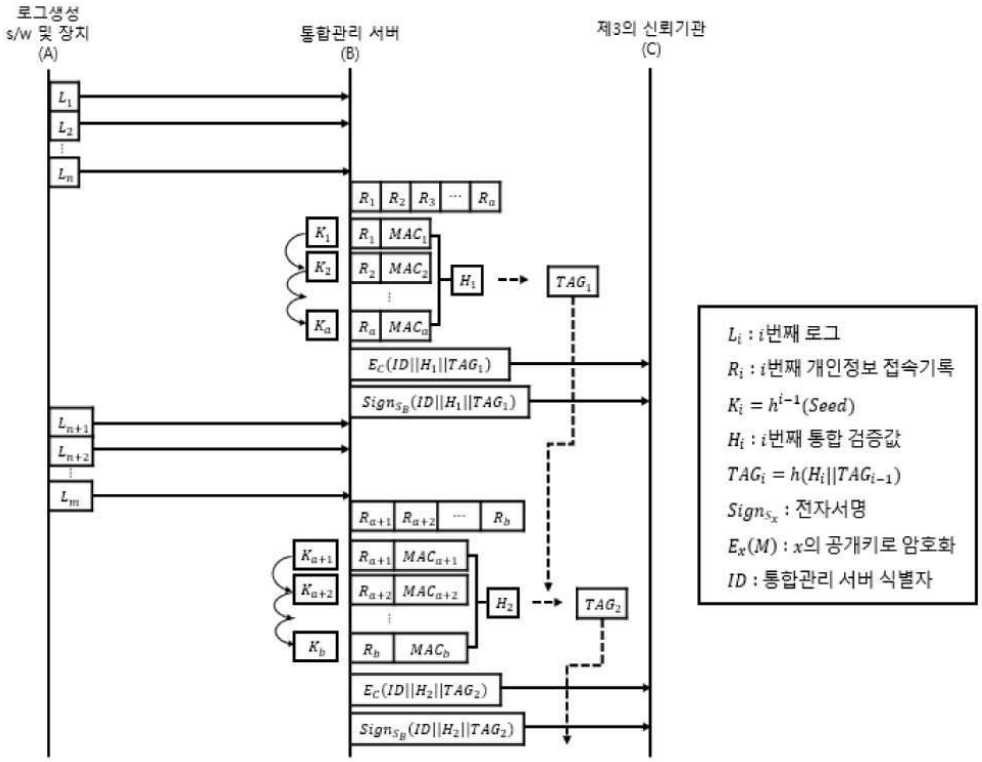
도면8


로그생성 S/W 및 장치
(A)
통합관리 서버
(B)
제3의 신뢰기관
(C)
P_B S_B
P_C S_C
P_C request
P_C
P_B
K_1 ID
K_1 request
$E_C(K_1||ID)$
$Sign_B(K_1||ID)$
P_x : x의 공개키
S_x : x의 개인키
K_1 : 초기 비밀키
$E_x(M)$: x의 공개키로 암호화
$Sign_x(M)$: x의 개인키로 전자서명

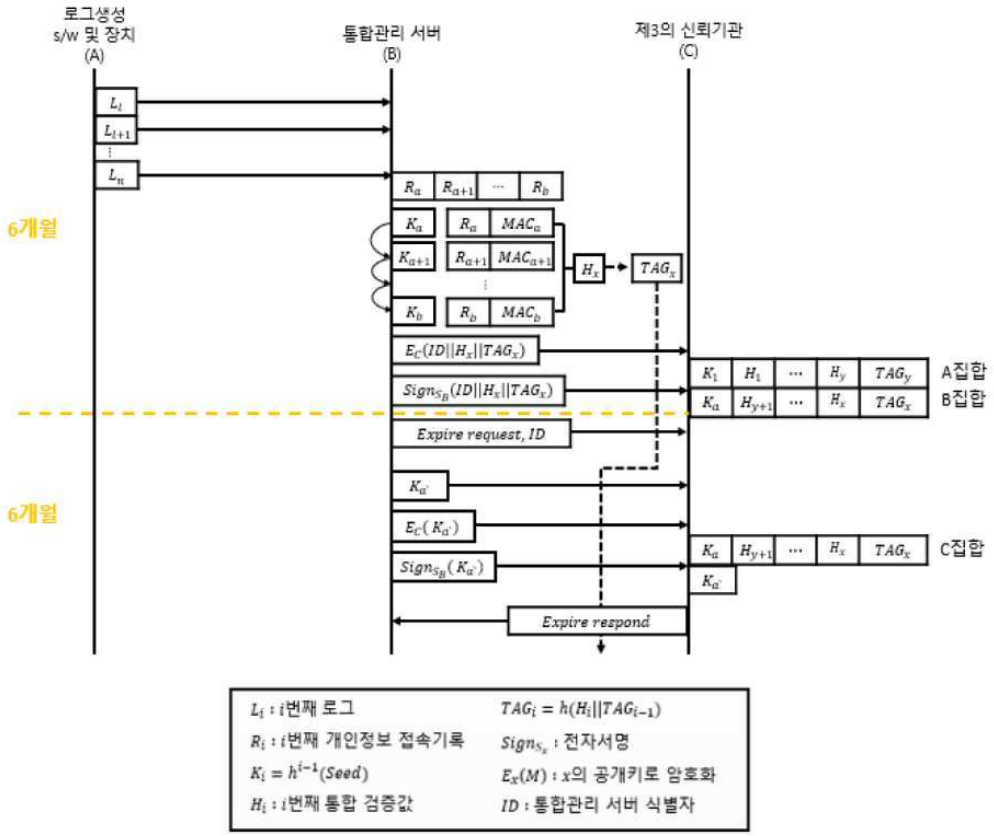
도면9


로그생성 s/w 및 장치
통합관리 서버
제3의 신뢰기관
① 로그생성
② 로그 전송
③ 개인정보 접속기록 추출 및 통합
④ 개별 검증값 생성
④ 통합 검증값 생성
⑤ 검증값 전송
⑥ 검증값 확인

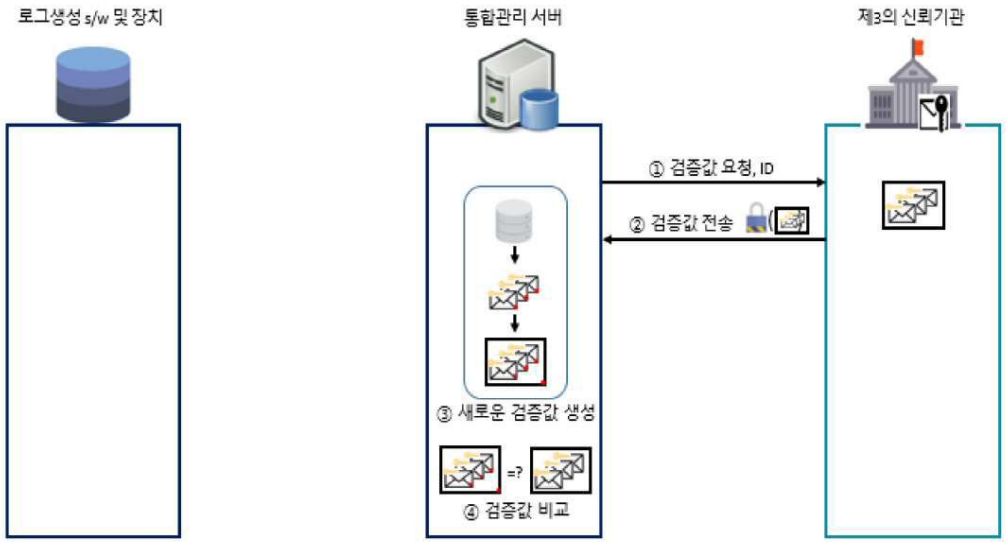
도면10


로그생성 s/w 및 장치 (A)
통합관리 서버 (B)
제3의 신뢰기관 (C)
L_i : i번째 로그
R_i : i번째 개인정보 접속기록
$K_i = h^{i-1}(Seed)$
H_i : i번째 통합 검증값
$TAG_i = h(H_i||TAG_{i-1})$
$Sign_{S_x}$: 전자서명
$E_x(M)$: x의 공개키로 암호화
ID : 통합관리 서버 식별자

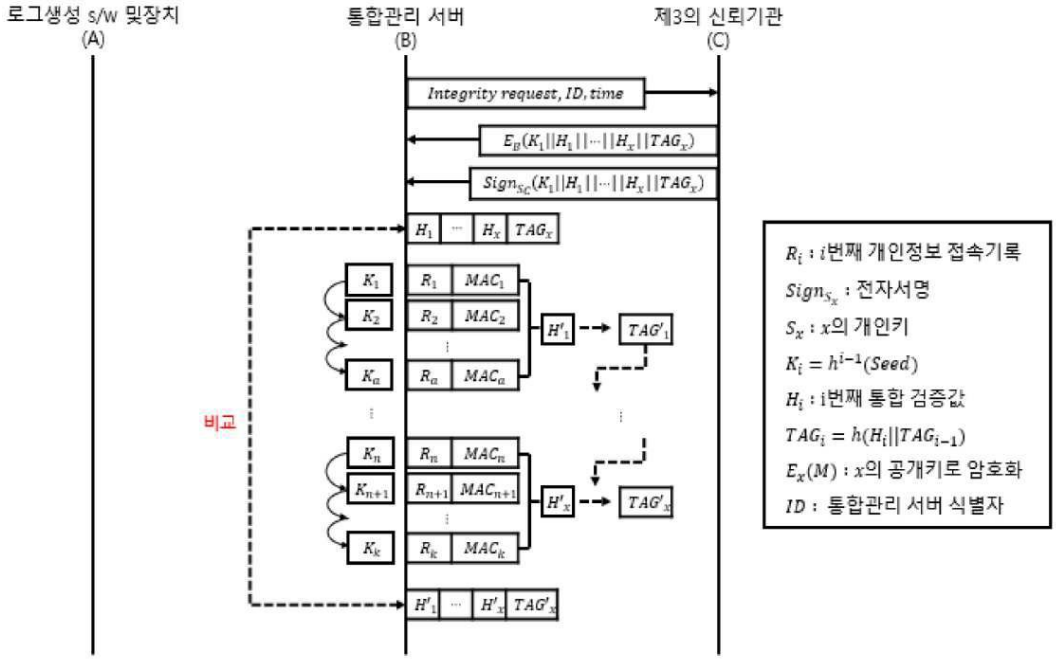
도면11


로그생성
s/w 및 장치
(A)
통합관리 서버
(B)
제3의 신뢰기관
(C)
6개월
6개월
A집합
B집합
C집합
Expire request, ID
Expire respond
L_i : i번째 로그
R_i : i번째 개인정보 접속기록
$K_i = h^{i-1}(Seed)$
H_i : i번째 통합 검증값
$TAG_i = h(H_i||TAG_{i-1})$
$Sign_{S_x}$: 전자서명
$E_x(M)$: x의 공개키로 암호화
ID : 통합관리 서버 식별자

도면12


로그생성 s/w 및 장치
통합관리 서버
제3의 신뢰기관
① 검증값 요청, ID
② 검증값 전송
③ 새로운 검증값 생성
=?
④ 검증값 비교

도면13


로그생성 s/w 및장치
(A)
통합관리 서버
(B)
제3의 신뢰기관
(C)
Integrity request, ID, time
$E_B(K_1||H_1||\cdots||H_x||TAG_x)$
$Sign_{S_C}(K_1||H_1||\cdots||H_x||TAG_x)$
H_1 ··· H_x TAG_x
K_1 K_2 K_n
R_1 MAC_1
R_2 MAC_2
R_n MAC_n
H'_1
TAG'_1
K_n K_{n+1} K_k
R_n MAC_n
R_{n+1} MAC_{n+1}
R_k MAC_k
H'_x
TAG'_x
비교
H'_1 ··· H'_x TAG'_x
R_i : i번째 개인정보 접속기록
$Sign_{S_x}$: 전자서명
S_x : x의 개인키
$K_i = h^{i-1}(Seed)$
H_i : i번째 통합 검증값
$TAG_i = h(H_i||TAG_{i-1})$
$E_x(M)$: x의 공개키로 암호화
ID : 통합관리 서버 식별자