(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2020년06월03일
(11) 등록번호 10-2116573
(24) 등록일자 2020년05월22일

(51) 국제특허분류(Int. Cl.)
G06F 21/56 (2013.01)
(52) CPC특허분류
G06F 21/56 (2013.01)
H04L 63/1425 (2013.01)
(21) 출원번호 10-2019-7012029
(22) 출원일자(국제) 2017년10월26일
심사청구일자 2019년10월31일
(85) 번역문제출일자 2019년04월25일
(65) 공개번호 10-2019-0067820
(43) 공개일자 2019년06월17일
(86) 국제출원번호 PCT/EP2017/077390
(87) 국제공개번호 WO 2018/077996
국제공개일자 2018년05월03일
(30) 우선권주장
15/336,387 2016년10월27일 미국(US)
(56) 선행기술조사문헌
KR1020160055826 A*
US20070240222 A1*
US20080209552 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
비트데펜더 아이피알 매니지먼트 엘티디
사이프러스 니코시아 1076 12 피시 크레온토스
(72) 발명자
하지마샨, 게오르게-플로린
루마니아 주데츠 알바, 547037, 코무나 룬카 무레술루이, 사트 룬카 무레술루이 엔알. 351
몬도크, 알렉산드라
루마니아 주데츨 클루지, 400607, 클루지-나포카, 피아타 마라스티 엔알. 3 에스씨. 아이, 이티. IV, 에이피. 17
포르타세, 라두-마리안
루마니아 주데츠 마라무레슈, 435700, 비세우 데 수스, 스트라다 드라고스 보다 엔알. 85에이
(74) 대리인
권영준

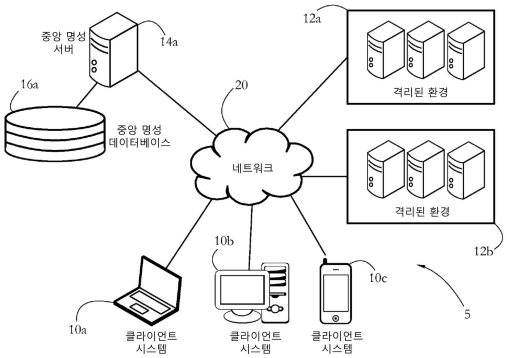
전체 청구항 수 : 총 21 항 심사관 : 정성훈

(54) 발명의 명칭 **컴퓨터 보안 작동을 최적화하기 위한 동적 명성 표시자**

(57) 요 약

설명된 시스템 및 방법은 바이러스, 웜, 및 스파이웨어와 같은 멀웨어로부터 컴퓨터 시스템을 보호할 수 있게 한다. 명성 관리자는 안티-멀웨어 엔진과 동시에 컴퓨터 시스템 상에서 실행된다. 명성 관리자는 동적 명성 표시자를 개별 구성 요소들(예를 들어, 메인 실행 파일 및 로딩된 라이브러리의 세트)의 고유한 조합으로 간주되는 각각의 실행가능한 엔터티에 연계시킨다. 명성 표시자는 개별 엔터티가 악성일 가능성을 나타낸다. 양성 엔터티의 명성은 시간에 따라 증가한다. 엔터티가 악성 활동을 나타낼 수도 있는 특정 활동을 수행한 경우, 개별 엔터티의 명성은 하락할 수 있다. 안티-멀웨어 엔진은 엔터티-특정적 프로토콜을 사용하여 악성에 대해 개별 타겟 엔터티를 스캔 및/또는 모니터링하고, 상기 프로토콜은 엔터티의 명성에 따라 변한다. 악성이 아닌 것으로 신뢰되는 엔터티들은 알 수 없는 또는 신뢰할 수 없는 엔터티들보다 더욱 완화된 프로토콜을 사용하여 분석될 수 있다.

대 표 도 - 도1



(52) CPC특허분류
H04L 63/145 (2013.01)
H04L 63/20 (2013.01)

명 세 서

청구범위

청구항 1

타겟 엔터티, 명성 관리자, 및 안티-멀웨어 엔진을 실행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 클라이언트 시스템으로서,

상기 명성 관리자는:

명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타내고,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내고,

상기 제2 명성 표시자를 결정하는 것에 응답으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하고,

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타내고,

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하도록 구성되고; 그리고

상기 안티-멀웨어 엔진은:

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하고,

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮고, 그리고

상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높도록 구성되며,

상기 제3 명성 표시자를 결정하는 것은 상기 타겟 엔터티가 상기 제1 활동 이전에 제2 활동을 수행하였는지 여부에 따라 결정된 양에 의해 상기 타겟 엔터티가 악성일 가능성을 증가시키는 것을 포함하는 것을 특징으로 하는 클라이언트 시스템.

청구항 2

제1항에 있어서,

상기 명성 관리자는:

상기 제2 또는 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제1 시간 간격 후속의 제2 시간 간격을 결정하고;

상기 제2 시간 간격을 결정하는 것에 응답으로, 상기 타겟 엔터티가 상기 제2 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고;

응답으로, 상기 타겟 엔터티가 상기 제2 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제4 명성 표시자를 결정하고, 상기 제4 명성 표시자는 상기 타겟 엔터티가 상기

제2 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내도록 추가적으로 구성되는 것을 특징으로 하는 클라이언트 시스템.

청구항 3

제1항에 있어서,

상기 제2 명성 표시자는 상기 타겟 엔터티의 개시(launch) 이후 경과된 시간에 따라 결정되는 것을 특징으로 하는 클라이언트 시스템.

청구항 4

제1항에 있어서,

상기 제1 시간 간격은 상기 타겟 엔터티의 개시 이후 경과된 시간에 따라 결정되는 것을 특징으로 하는 클라이언트 시스템.

청구항 5

제1항에 있어서,

상기 제1 시간 간격은 상기 제1 명성 표시자에 따라 결정되는 것을 특징으로 하는 클라이언트 시스템.

청구항 6

제1항에 있어서,

상기 제1 시간 간격은 상기 타겟 엔터티가 상기 제1 시간 간격 이전에 미리 정해진 활동들의 세트의 제2 활동을 수행하였는지 여부에 따라 결정되는 것을 특징으로 하는 클라이언트 시스템.

청구항 7

제1항에 있어서,

상기 제2 명성 표시자를 결정하는 것은 상기 타겟 엔터티의 개시 이후로 경과된 시간에 따라 결정된 양에 의해 상기 타겟 엔터티가 악성일 가능성을 감소시키는 것을 포함하는 것을 특징으로 하는 클라이언트 시스템.

청구항 8

제1항에 있어서,

상기 제3 명성 표시자를 결정하는 것은 상기 제1 활동의 유형에 따라 결정된 양에 의해 상기 타겟 엔터티가 악성일 가능성을 증가시키는 것을 포함하는 것을 특징으로 하는 클라이언트 시스템.

청구항 9

삭제

청구항 10

타겟 엔터티, 명성 관리자, 및 안티-멀웨어 엔진을 실행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 클라이언트 시스템으로서,

상기 명성 관리자는:

명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타내고,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성

표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내고,

상기 제2 명성 표시자를 결정하는 것에 응답으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하고,

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타내고,

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하도록 구성되고; 그리고

상기 안티-멀웨어 엔진은:

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하고,

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮고, 그리고

상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높도록 구성되고,

상기 명성 관리자는:

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제2 활동을 수행하였는지 여부를 결정하고; 그리고

응답으로, 상기 타겟 엔터티가 상기 제2 활동을 수행한 경우, 상기 타겟 엔터티의 제4 명성 표시자를 결정하고, 상기 제4 명성 표시자는 상기 타겟 엔터티가 상기 제3 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타내도록 추가적으로 구성되는 것을 특징으로 하는 클라이언트 시스템.

청구항 11

타겟 엔터티, 명성 관리자, 및 안티-멀웨어 엔진을 실행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 클라이언트 시스템으로서,

상기 명성 관리자는:

명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타내고,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내고,

상기 제2 명성 표시자를 결정하는 것에 응답으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하고,

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타내고,

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하도록 구성되고; 그리고

상기 안티-멀웨어 엔진은:

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토

콜을 채용하고,

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮고, 그리고

상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높도록 구성되고,

상기 명성 관리자는:

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 클라이언트 시스템 상에서 실행되는 다른 엔터티의 제4 명성 표시자를 결정하고, 상기 다른 엔터티는 상기 타겟 엔터티의 구성 요소를 포함하도록 추가적으로 구성되는 것을 특징으로 하는 클라이언트 시스템.

청구항 12

타겟 엔터티, 명성 관리자, 및 안티-멀웨어 엔진을 실행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 클라이언트 시스템으로서,

상기 명성 관리자는:

명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타내고,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내고,

상기 제2 명성 표시자를 결정하는 것에 응답으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하고,

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타내고,

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하도록 구성되고; 그리고

상기 안티-멀웨어 엔진은:

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하고,

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮고, 그리고

상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높도록 구성되고,

상기 제1 활동은 상기 타겟 엔터티가 상기 클라이언트 시스템 상에서 실행되는 다른 엔터티로 코드의 섹션을 삽입하는 것을 포함하고, 상기 명성 관리자는, 상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 클라이언트 시스템 상에서 실행되는 상기 다른 엔터티의 제4 명성 표시자를 결정하도록 추가적으로 구성되고, 상기 제4 명성 표시자는 상기 다른 엔터티가 상기 타겟 엔터티 만큼 악성일 가능성이 있다는 것을 나타내는 것을 특징으로 하는 클라이언트 시스템.

청구항 13

복수의 클라이언트 시스템들과 명성 관리 트랜잭션을 수행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 서버 컴퓨터 시스템으로서,

상기 명성 관리 트랜잭션은:

상기 복수의 클라이언트 시스템들의 클라이언트 시스템으로부터 수신한 요청에 응답하여, 엔터티 명성 데이터베이스로부터 타겟 엔터티의 제1 명성 표시자를 검색하는 것(상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타냄);

상기 제1 명성 표시자를 검색하는 것에 응답으로, 상기 제1 명성 표시자를 상기 클라이언트 시스템으로 전송하는 것;

상기 제1 명성 표시자를 전송하는 것에 응답으로, 상기 클라이언트 시스템으로부터 상기 타겟 엔터티의 제2 명성 표시자를 수신하는 것;

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 제1 및 제2 명성 표시자들을 비교하는 것;

응답으로, 상기 제2 명성 표시자가, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하는 경우, 상기 제2 명성 표시자를 상기 복수의 클라이언트 시스템들로부터 수신한 명성 표시자들의 컬렉션에 추가하는 것(상기 컬렉션의 모든 구성원들은 상기 타겟 엔터티의 인스턴스를 위해 결정됨);

상기 컬렉션에 상기 제2 명성 표시자를 추가하는 것에 응답으로, 명성 업데이트 조건이 만족되는지 여부를 결정하는 것; 및

응답으로, 상기 업데이트 조건이 만족되는 경우, 상기 명성 데이터베이스 내 상기 제1 명성 표시자를 상기 컬렉션에 따라 결정된 업데이트 된 명성 표시자로 교체하는 것;을 포함하고,

그리고 상기 제2 명성 표시자를 결정하는 것은,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하기 위해 상기 제2 명성 표시자를 형성하고, 그리고

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 높다고 표시하기 위해 상기 제2 명성 표시자를 형성하기 위해 클라이언트 시스템을 채용하는 것을 포함하며,

상기 업데이트 조건이 만족되는지 여부를 결정하는 것은 상기 컬렉션에 제1 구성원을 추가한 후 경과한 시간을 결정하는 것을 포함하는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 14

삭제

청구항 15

복수의 클라이언트 시스템들과 명성 관리 트랜잭션을 수행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 서버 컴퓨터 시스템으로서,

상기 명성 관리 트랜잭션은:

상기 복수의 클라이언트 시스템들의 클라이언트 시스템으로부터 수신한 요청에 응답하여, 엔터티 명성 데이터베이스로부터 타겟 엔터티의 제1 명성 표시자를 검색하는 것(상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타냄);

상기 제1 명성 표시자를 검색하는 것에 응답으로, 상기 제1 명성 표시자를 상기 클라이언트 시스템으로 전송하는 것;

상기 제1 명성 표시자를 전송하는 것에 응답으로, 상기 클라이언트 시스템으로부터 상기 타겟 엔터티의 제2 명성 표시자를 수신하는 것;

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 제1 및 제2 명성 표시자들을 비교하는 것;

응답으로, 상기 제2 명성 표시자가, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하는 경우, 상기 제2 명성 표시자를 상기 복수의 클라이언트 시스템들로부터 수신한 명성 표시자들의 컬렉션에 추가하는 것(상기 컬렉션의 모든 구성원들은 상기 타겟 엔터티의 인스턴스를 위해 결정됨);

상기 컬렉션에 상기 제2 명성 표시자를 추가하는 것에 응답으로, 명성 업데이트 조건이 만족되는지 여부를 결정하는 것; 및

응답으로, 상기 업데이트 조건이 만족되는 경우, 상기 명성 데이터베이스 내 상기 제1 명성 표시자를 상기 컬렉션에 따라 결정된 업데이트 된 명성 표시자로 교체하는 것;을 포함하고,

그리고 상기 제2 명성 표시자를 결정하는 것은,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하기 위해 상기 제2 명성 표시자를 형성하고, 그리고

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 높다고 표시하기 위해 상기 제2 명성 표시자를 형성하기 위해 클라이언트 시스템을 채용하는 것을 포함하며,

상기 업데이트 조건이 만족되는지 여부를 결정하는 것은 상기 컬렉션의 구성원들의 개수를 결정하는 것을 포함하는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 16

복수의 클라이언트 시스템들과 명성 관리 트랜잭션을 수행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 서버 컴퓨터 시스템으로서,

상기 명성 관리 트랜잭션은:

상기 복수의 클라이언트 시스템들의 클라이언트 시스템으로부터 수신한 요청에 응답하여, 엔터티 명성 데이터베이스로부터 타겟 엔터티의 제1 명성 표시자를 검색하는 것(상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타냄);

상기 제1 명성 표시자를 검색하는 것에 응답으로, 상기 제1 명성 표시자를 상기 클라이언트 시스템으로 전송하는 것;

상기 제1 명성 표시자를 전송하는 것에 응답으로, 상기 클라이언트 시스템으로부터 상기 타겟 엔터티의 제2 명성 표시자를 수신하는 것;

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 제1 및 제2 명성 표시자들을 비교하는 것;

응답으로, 상기 제2 명성 표시자가, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하는 경우, 상기 제2 명성 표시자를 상기 복수의 클라이언트 시스템들로부터 수신한 명성 표시자들의 컬렉션에 추가하는 것(상기 컬렉션의 모든 구성원들은 상기 타겟 엔터티의 인스턴스를 위해 결정됨);

상기 컬렉션에 상기 제2 명성 표시자를 추가하는 것에 응답으로, 명성 업데이트 조건이 만족되는지 여부를 결정하는 것; 및

응답으로, 상기 업데이트 조건이 만족되는 경우, 상기 명성 데이터베이스 내 상기 제1 명성 표시자를 상기 컬렉션에 따라 결정된 업데이트 된 명성 표시자로 교체하는 것;을 포함하고,

그리고 상기 제2 명성 표시자를 결정하는 것은,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하기 위해 상기

제2 명성 표시자를 형성하고, 그리고

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 높다고 표시하기 위해 상기 제2 명성 표시자를 형성하기 위해 클라이언트 시스템을 채용하는 것을 포함하며,

상기 업데이트 명성 표시자를 결정하는 것은 상기 타겟 엔터티가 상기 컬렉션의 모든 구성원들에 대해 가장 높은 악성 가능성을 나타내도록 상기 업데이트 명성 표시자를 형성하는 것을 포함하는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 17

복수의 클라이언트 시스템들과 명성 관리 트랜잭션을 수행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하는 서버 컴퓨터 시스템으로서,

상기 명성 관리 트랜잭션은:

상기 복수의 클라이언트 시스템들의 클라이언트 시스템으로부터 수신한 요청에 응답하여, 엔터티 명성 데이터베이스로부터 타겟 엔터티의 제1 명성 표시자를 검색하는 것(상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타냄);

상기 제1 명성 표시자를 검색하는 것에 응답으로, 상기 제1 명성 표시자를 상기 클라이언트 시스템으로 전송하는 것;

상기 제1 명성 표시자를 전송하는 것에 응답으로, 상기 클라이언트 시스템으로부터 상기 타겟 엔터티의 제2 명성 표시자를 수신하는 것;

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 제1 및 제2 명성 표시자들을 비교하는 것;

응답으로, 상기 제2 명성 표시자가, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하는 경우, 상기 제2 명성 표시자를 상기 복수의 클라이언트 시스템들로부터 수신한 명성 표시자들의 컬렉션에 추가하는 것(상기 컬렉션의 모든 구성원들은 상기 타겟 엔터티의 인스턴스를 위해 결정됨);

상기 컬렉션에 상기 제2 명성 표시자를 추가하는 것에 응답으로, 명성 업데이트 조건이 만족되는지 여부를 결정하는 것; 및

응답으로, 상기 업데이트 조건이 만족되는 경우, 상기 명성 데이터베이스 내 상기 제1 명성 표시자를 상기 컬렉션에 따라 결정된 업데이트 된 명성 표시자로 교체하는 것;을 포함하고,

그리고 상기 제2 명성 표시자를 결정하는 것은,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하기 위해 상기 제2 명성 표시자를 형성하고, 그리고

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 높다고 표시하기 위해 상기 제2 명성 표시자를 형성하기 위해 클라이언트 시스템을 채용하는 것을 포함하며,

상기 제2 명성 표시자를 결정하는 것은, 상기 타겟 엔터티가 제3 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 표시하기 위해 상기 제2 명성 표시자를 형성하는 것을 포함하고, 상기 제3 명성 표시자를 결정하는 것은:

상기 제2 명성 표시자를 결정하기 위한 준비로서, 상기 제1 시간 간격 이전에 제2 시간 간격을 결정하고;

상기 제2 시간 간격을 결정하는 것에 응답으로, 상기 타겟 엔터티가 상기 제2 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고;

응답으로, 상기 제2 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟

엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 표시하기 위해 상기 제3 명성 표시자를 형성하기 위해 클라이언트 시스템을 채용하는 것을 포함하는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 18

제13항에 있어서,

상기 제2 명성 표시자는 상기 타겟 엔터티의 개시 이후 경과된 시간에 따라 결정되는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 19

제13항에 있어서,

상기 제1 시간 간격은 상기 타겟 엔터티의 개시 이후 경과된 시간에 따라 결정되는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 20

제13항에 있어서,

상기 제1 시간 간격은 상기 제1 명성 표시자에 따라 결정되는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 21

제13항에 있어서,

상기 제1 시간 간격은 상기 타겟 엔터티가 상기 제1 시간 간격 이전에 미리 정해진 활동들의 세트의 제2 활동을 수행하였는지 여부에 따라 결정되는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 22

제13항에 있어서,

상기 제2 명성 표시자를 결정하는 것은 상기 타겟 엔터티의 개시 이후로 경과된 시간에 따라 결정된 양에 의해 상기 타겟 엔터티가 악성일 가능성을 감소시키는 것을 포함하는 것을 특징으로 하는 서버 컴퓨터 시스템.

청구항 23

클라이언트 시스템의 하드웨어 프로세서에 의하여 실행될 때, 상기 클라이언트 시스템으로 하여금 명성 관리자 및 안티-멀웨어 엔진을 형성하도록 하는 명령들의 세트를 저장하는 비-일시적 컴퓨터 판독가능 매체(non-transitory computer-readable medium)로서,

상기 클라이언트 시스템은 타겟 엔터티를 실행하도록 구성되고;

상기 명성 관리자는:

명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타내고,

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하고,

상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타내고,

상기 제2 명성 표시자를 결정하는 것에 응답으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하고,

상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성

일 가능성이 높다는 것을 나타내고,

상기 제3 명성 표시자를 결정하는 것에 응답으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송하도록 구성되고; 그리고

상기 안티-멀웨어 엔진은:

상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하고,

상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮고, 그리고

상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높도록 구성되며,

상기 제3 명성 표시자를 결정하는 것은 상기 타겟 엔터티가 상기 제1 활동 이전에 제2 활동을 수행하였는지 여부에 따라 결정된 양에 의해 상기 타겟 엔터티가 악성일 가능성을 증가시키는 것을 포함하는 것을 특징으로 하는 비-일시적 컴퓨터 판독가능 매체.

발명의 설명

기 술 분 야

[0001] 본 발명은 악성 소프트웨어로부터 컴퓨터 시스템을 보호하기 위한 시스템 및 방법에 대한 것이다.

배 경 기 술

[0002] 멀웨어로도 알려진 악성 소프트웨어는 세계적으로 많은 수의 컴퓨터 시스템에 영향을 주고 있다. 멀웨어는 컴퓨터 바이러스, 웜, 루트킷(rootkit) 및 스파이웨어와 같은 많은 형태로, 수백만의 컴퓨터 사용자에게 심각한 위협이 되고 있으며, 무엇보다도 데이터 및 민감한 정보의 손실, 프라이버시 침해, 신원 도용, 및 생산성 손실에 있어 이들을 취약하게 하고 있다.

[0003] 보안 소프트웨어는 사용자의 컴퓨터 시스템을 감염시키는 멀웨어를 탐지하기 위하여, 그러한 멀웨어를 제거 및/또는 불능화시키기 위하여 사용될 수 있다. 여러 멀웨어-탐지 기술들이 본 기술분야에서 알려져 있다. 일부는 상기 멀웨어 에이전트의 코드의 프래그먼트(fragment)를 멀웨어를 나타내는 시그니쳐들의 라이브러리에 매칭하는 것에 의존하는, 콘텐트 기반형이다. 다른 종래의 기술들, 흔히 행위기반(behavioral)으로 알려져 있는 것들은, 멀웨어 에이전트의 의심스럽거나 또는 멀웨어를 나타내는 활동들의 세트를 탐지한다.

[0004] 보안 소프트웨어는 종종 성능 및 사용자 경험에 대해 측정가능한 영향력을 가지는, 사용자의 컴퓨터 시스템에 상당한 컴퓨팅 부담(computational burden)을 부가할 수 있다. 지속적인 악성 소프트웨어의 확산은 시그니쳐 데이터베이스(signature database)의 크기뿐만 아니라 멀웨어 탐지 루틴의 복잡성을 더욱 증가시킨다. 컴퓨팅 비용을 줄이기 위하여 보안 소프트웨어는 여러 최적화 절차를 병합할 수 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0005] 일 태양에 따르면, 클라이언트 시스템은 타겟 엔터티, 명성 관리자, 및 안티-멀웨어 엔진을 실행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함한다. 상기 명성 관리자는, 명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하도록 구성되고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타낸다. 상기 명성 관리자는, 상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하도록 추가적으로 구성된다. 상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 명성 관리자는 상기 타겟 엔터티의 제2 명성

표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타낸다. 상기 명성 관리자는 추가적으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송한다. 상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 명성 관리자는 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타낸다. 상기 명성 관리자는 추가적으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송한다. 상기 안티-멀웨어 엔진은, 상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하도록 구성된다. 상기 안티-멀웨어 엔진은 추가적으로, 상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하도록 구성되고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮다. 상기 안티-멀웨어 엔진은 추가적으로, 상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하도록 구성되고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높다.

[0006] 다른 태양에 따르면, 서버 컴퓨터 시스템은 복수의 클라이언트 시스템들과 명성 관리 트랜잭션을 수행하도록 구성된 적어도 하나의 하드웨어 프로세서를 포함하고, 여기서 명성 관리 트랜잭션은, 상기 복수의 클라이언트 시스템들의 클라이언트 시스템으로부터 수신한 요청에 응답하여, 엔터티 명성 데이터베이스로부터 타겟 엔터티의 제1 명성 표시자를 검색하는 것을 포함하고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타낸다. 상기 트랜잭션은 추가적으로, 상기 제1 명성 표시자를 검색하는 것에 응답으로, 상기 제1 명성 표시자를 상기 클라이언트 시스템으로 전송하는 것 및 상기 제1 명성 표시자를 전송하는 것에 응답으로, 상기 클라이언트 시스템으로부터 상기 타겟 엔터티의 제2 명성 표시자를 수신하는 것을 포함한다. 상기 트랜잭션은 추가적으로, 상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 제1 및 제2 명성 표시자들을 비교하는 것을 포함한다. 응답으로, 상기 제2 명성 표시자가, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하는 경우, 상기 트랜잭션은 추가적으로, 상기 제2 명성 표시자를 상기 복수의 클라이언트 시스템들로부터 수신한 명성 표시자들의 컬렉션에 추가하는 것을 포함하고, 상기 컬렉션의 모든 구성원들은 상기 타겟 엔터티의 인스턴스를 위해 결정된다. 상기 트랜잭션은 추가적으로, 상기 컬렉션에 상기 제2 명성 표시자를 추가하는 것에 응답으로, 명성 업데이트 조건이 만족되는지 여부를 결정하는 것, 및 응답으로, 상기 업데이트 조건이 만족되는 경우, 상기 명성 데이터베이스 내 상기 제1 명성 표시자를 상기 컬렉션에 따라 결정된 업데이트 된 명성 표시자로 교체하는 것을 포함한다. 상기 제2 명성 표시자를 결정하는 것은, 상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하기 위해 클라이언트 시스템을 채용하는 것을 포함한다. 상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 제2 명성 표시자를 결정하는 것은 추가적으로, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 낮다고 표시하기 위해 상기 제2 명성 표시자를 형성하는 것, 그리고 상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 제1 명성 표시자에 의해 표시된 것보다 상기 타겟 엔터티가 악성일 가능성이 높다고 표시하기 위해 상기 제2 명성 표시자를 형성하는 것을 포함한다.

[0007] 또 다른 태양에 따르면, 비-일시적 컴퓨터 판독가능 매체(non-transitory computer-readable medium)는 클라이언트 시스템의 하드웨어 프로세서에 의하여 실행될 때, 상기 클라이언트 시스템으로 하여금 명성 관리자 및 안티-멀웨어 엔진을 형성하도록 하는 명령들의 세트를 저장한다. 상기 클라이언트 시스템은 타겟 엔터티를 실행한다. 상기 명성 관리자는, 명성 서버로부터 타겟 엔터티의 제1 명성 표시자를 수신하는 것에 응답으로, 상기 명성 표시자를 상기 안티-멀웨어 엔진으로 전송하도록 구성되고, 상기 제1 명성 표시자는 상기 타겟 엔터티가 악성일 가능성을 나타낸다. 상기 명성 관리자는 추가적으로, 상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하였는지 여부를 결정하도록 구성된다. 상기 타겟 엔터티가 상기 제1 시간 간격 동안 미리 정해진 활동들의 세트 중 임의의 것을 수행하지 않은 경우, 상기 명성 관리자는 상기 타겟 엔터티의 제2 명성 표시자를 결정하고, 상기 제2 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 낮다는 것을 나타낸다. 상기 명성 관리자는 추가적으로, 상기 제2 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송한다. 상기 타겟 엔터티가 미리 정해진 활동들의 세트의 제1 활동을 수행한 경우, 상기 명성 관리자는 상기 타겟 엔터티의 제3 명성 표시자를 결정하고, 상기 제3 명성 표시자는 상기 타겟 엔터티가 상기 제1 명성 표시자에 의해 표시된 것보다 악성일 가능성이 높다는 것을 나타낸다. 상기 명성 관리자는 추가적으로, 상기 제3 명성 표시자를 상기 안티-멀웨어 엔진으로 그리고 상기 명성 서버로 전송한다. 상기 안티-멀웨어 엔진은, 상기 제1 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제1 프로토콜을 채용하

도록 구성된다. 상기 안티-멀웨어 엔진은 추가적으로, 상기 제2 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제2 프로토콜을 채용하도록 구성되고, 상기 제2 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 낮다. 상기 안티-멀웨어 엔진은 추가적으로, 상기 제3 명성 표시자를 수신하는 것에 응답으로, 상기 타겟 엔터티가 악성인지를 결정하기 위하여 제3 프로토콜을 채용하도록 구성되고, 상기 제3 프로토콜은 상기 제1 프로토콜보다 컴퓨팅 비용이 높다.

도면의 간단한 설명

[0008] 본 발명의 전술한 태양들 및 장점은 후술하는 상세한 설명 및 도면을 참조로 이해하면 더욱 잘 이해될 것이다.

도 1은 본 발명의 일부 실시예에 따른 복수의 클라이언트 시스템들과 명성 서버를 포함하는 예시적인 안티-멀웨어 시스템을 보여주는 도면.

도 2는 본 발명의 일부 실시예에 따른, 컴퓨터 보안 위협으로부터 보호된 기업 인트라넷과 같은 격리된 환경의 예시적인 상세도를 보여주는 도면.

도 3은 본 발명의 일부 실시예에 따른 예시적인 명성 데이터베이스 엔트리를 보여주는 도면.

도 4a는 본 발명의 일부 실시예에 따른 클라이언트 시스템의 예시적인 하드웨어 구성을 보여주는 도면.

도 4b는 본 발명의 일부 실시예에 따른 명성 서버의 예시적인 하드웨어 구성을 보여주는 도면.

도 5는 본 발명의 일부 실시예에 따른 컴퓨터 보안 위협으로부터 클라이언트 시스템을 보호하도록 구성된 보안 어플리케이션을 포함하는 클라이언트 시스템에서 실행되는 소프트웨어 객체들의 예시적인 세트를 보여주는 도면.

도 6은 본 발명의 일부 실시예에 따른 보안 어플리케이션의 예시적인 구성 요소들을 보여주는 도면.

도 7은 본 발명의 일부 실시예에 따른 보안 어플리케이션의 안티-멀웨어 엔진 구성 요소와 명성 관리자 구성 요소 사이의 예시적인 데이터 교환을 보여주는 도면.

도 8은 본 발명의 일부 실시예에 따른 클라이언트 시스템과 명성 서버 사이의 예시적인 데이터 교환을 보여주는 도면.

도 9는 본 발명의 일부 실시예에 따른 실행가능한 엔터티의 핑거프린트(fingerprint)의 예시적인 구성 요소를 보여주는 도면.

도 10은 본 발명의 일부 실시예에 따른 실행가능한 엔터티들의 예시적인 세트 및 초집합(superset)을 보여주는 도면.

도 11은 본 발명의 일부 실시예에 따른, 클라이언트 시스템에서 실행되는 실행가능한 엔터티에 관련된 예시적인 데이터 구조를 보여주는 도면.

도 12a는 본 발명의 일부 실시예에 따른 보안 어플리케이션의 명성 관리자 구성 요소에 의해 수행되는 단계들의 예시적인 시퀀스를 보여주는 도면.

도 12b는 본 발명의 일부 실시예에 따른 도 11a의 단계들의 예시적인 시퀀스의 연속을 보여주는 도면.

도 12c는 본 발명의 일부 실시예에 따른 도 11a의 단계들의 예시적인 시퀀스의 다른 연속을 보여주는 도면.

도 12d는 본 발명의 일부 실시예에 따른 도 11a의 단계들의 예시적인 시퀀스의 또 다른 연속을 보여주는 도면.

도 13은 본 발명의 일부 실시예에 따른 명성 표시자의 예시적인 시간적 발달을 보여주는 도면.

도 14는 본 발명의 일부 실시예에 따른 보안 어플리케이션의 안티-멀웨어 엔진 구성 요소에 의해 수행되는 단계들의 예시적인 시퀀스를 보여주는 도면.

도 15는 본 발명의 일부 실시예에 따른 명성 서버에 의해 수행되는 단계들의 예시적인 시퀀스를 보여주는 도면.

발명을 실시하기 위한 구체적인 내용

[0009] 이하의 설명에서, 구조들 사이에서 언급된 모든 연결들은 직접적인 동작 연결들 또는 매개 구조들을 통한 간접적인 동작 연결들일 수 있는 것으로 이해된다. 구성 요소들의 세트는 하나 이상의 구성 요소를 포함한다. 구성

요소의 임의의 열거는 적어도 하나의 구성 요소를 언급하는 것으로 이해된다. 복수의 구성 요소는 적어도 2개의 구성 요소를 포함한다. 달리 요구되지 않는다면, 기술된 어떠한 방법 단계들도 설명된 특정 순서로 반드시 실행될 필요는 없다. 제2 구성 요소로부터 유도되는 제1 구성 요소(예컨대, 데이터)는 제2 구성 요소와 동일한 제1 구성 요소는 물론, 제2 구성 요소 그리고 선택적으로는 다른 데이터를 처리하는 것에 의해 생성된 제1 구성 요소를 포함한다. 파라미터에 따라 결정 또는 판정하는 것은 파라미터에 따라 그리고 선택적으로는 다른 데이터에 따라 결정 또는 판정하는 것을 포함한다. 달리 구체화되지 않는다면, 일부 수량/데이터의 표시자는 수량/데이터 그 자체, 또는 수량/데이터 그 자체와 상이한 표시자일 수 있다. 컴퓨터 보안은 데이터 및/또는 하드웨어로의 의도되지 않았거나 인가받지 않은 접근에 대해서, 데이터 및/또는 하드웨어의 의도되지 않았거나 인가받지 않은 수정에 대해서, 그리고 데이터 및/또는 하드웨어의 파괴에 대해서 사용자와 장치를 보호하는 것을 포함한다. 컴퓨터 프로그램은 과업을 수행하는 프로세서 명령들의 시퀀스이다. 본 발명의 일부 실시예들에서 설명되는 컴퓨터 프로그램들은 독립형 소프트웨어 개체들 또는 다른 컴퓨터 프로그램들의 서브-개체들(예를 들어, 서브루틴들, 라이브러리들)일 수 있다. 달리 특정되지 않는다면, 프로세스는 별도의 메모리 공간 및 적어도 실행 쓰레드(execution thread)를 갖는 컴퓨터 프로그램의 인스턴스를 나타내고, 상기 메모리 공간은 프로세서 명령들의 세트의 인코딩(예를 들어, 머신 코드)을 저장한다. 달리 구체화되지 않는다면, 해시(hash)는 해시 함수(hash function)의 출력이다. 달리 구체화되지 않는다면, 해시 함수는 다양한 길이의 기호들(예를 들어, 문자, 비트)의 시퀀스를 고정된 길이의 비트 스트링으로 매핑하는 수학적 변환이다. 컴퓨터 판독 가능 매체는 자성, 광, 및 반도체 저장 매체와 같은 비-일시적 매체(non-transitory medium)(예컨대, 하드 드라이브, 광 디스크, 플래시 메모리, DRAM)는 물론, 전도성 케이블 및 파이버 옵틱 링크와 같은 통신 링크들을 포함한다. 일부 실시예들에 따르면, 본 발명은, 그 중에서도, 본원에 설명된 방법들을 수행하기 위해 프로그래밍된 하드웨어(예컨대, 하나 이상의 프로세서)는 물론, 본원에서 설명된 방법들을 수행하기 위해 명령들을 인코딩하는 컴퓨터-판독 가능 매체를 포함하는 컴퓨터 시스템을 제공한다.

[0010] 후술하는 설명은 본 발명의 실시예들을 예시적으로 설명하는 것이며, 반드시 제한적인 것은 아니다.

[0011] 도 1은 본 발명의 일부 실시예에 따른 예시적 컴퓨터 보안 시스템(5)을 보여준다. 시스템(5)은 통신 네트워크(20)로 연결된, 클라이언트 시스템(10a-c)들의 세트와 중앙 명성 서버(14a)를 포함한다. 중앙 명성 서버(14a)는 중앙 명성 데이터베이스(16a)에 추가적으로 통신가능하게 연결될 수 있다. 네트워크(20)는 인터넷과 같은 광역 네트워크일 수 있고, 한편 네트워크(20)의 일부는 또한 LAN(local area network)을 포함할 수 있다.

[0012] 시스템(5)은 네트워크(20)에 연결된 격리된 환경(isolated environment)(12a-b)들의 세트를 추가적으로 포함한다. 격리된 환경은, 예를 들어 기업 인트라넷(company Intranet)을 나타낼 수 있다. 환경(12a-b)들은 방화벽(firewall) 및/또는 다른 주변 방어 수단에 의해 네트워크(20)의 나머지 부분으로부터 분리될 수 있다. 도 2는 모두 로컬 네트워크(120)에 연결된 로컬 명성 서버(14b) 및 클라이언트 시스템(10d-e)들의 세트를 포함하는, 이러한 격리된 환경(12)을 도시한다. 네트워크(120)는 예를 들어, LAN(local area network)을 나타낼 수 있다. 일부 실시예들에서, 격리된 환경(12)은 로컬 명성 서버 (14b)에 통신가능하게 연결된 환경 특정 로컬 명성 데이터베이스(environment-specific local reputation database)(16b)를 추가적으로 포함할 수 있다.

[0013] 클라이언트 시스템(10a-e)들은 본 발명의 일부 실시예에 따라 컴퓨터 보안 위협으로부터 보호되는 최종 사용자 컴퓨터 시스템을 나타낸다. 예시적인 클라이언트 시스템(10a-e)들은, 태블릿 개인용 컴퓨터, 모바일 전화기, 휴대 정보 단말기(personal digital assistant, PDA), 웨어러블 컴퓨팅 장치(예컨대, 스마트워치)와 같은 개인용 컴퓨터, 모바일 컴퓨팅 및/또는 전기통신 장치, TV 또는 음악 플레이어와 같은 가정 기기, 또는 프로세서 및 메모리를 가지는 임의의 다른 전자 장치를 포함한다. 클라이언트 시스템(10a-e)들은 컴퓨터 보안 회사의 개별 고객들을 나타낼 수 있고, 여러 클라이언트 시스템들이 동일한 고객에 속할 수 있다.

[0014] 클라이언트 시스템(10a-e)들은 컴퓨터 보안 작동의 효율을 증가시키기 위해 명성 데이터를 사용할 수 있다. 일부 실시예들에서, 명성 서버(14a-b)들은 클라이언트 시스템(10a-e)들의 요청에 의해 명성 데이터를 처리하는데, 예를 들면, 명성 데이터베이스(16a-b)로/로부터 명성 데이터를 저장하고 선별적으로 검색하고, 그리고 그러한 데이터를 요청한 클라이언트 시스템에 전송한다. 이러한 트랜잭션의 상세한 내용은 이하에서 설명된다.

[0015] 명성 데이터베이스(16a-b)들은 다양한 실행가능한 엔터티들(어플리케이션, 운영 시스템의 구성 요소, 프로세스, 라이브러리, 스크립트 등)과 연관된 명성 데이터를 저장하도록 구성될 수 있다. 명성 데이터는 복수의 엔트리로서 저장될 수 있으며, 각각의 엔트리는 특유의 실행가능한 엔터티에 대응한다. 도 3은 실행가능한 엔터티의 식별 토큰(이하에서는 엔터티 핑거프린트(70)로 칭함) 및 개별 엔터티가 악성일 확률을 나타내는 명성 표시자(60)를 포함하는 예시적인 명성 데이터베이스 엔트리(17)를 보여준다. 각각의 명성 데이터베이스 엔트리는 표시자

(60)가 생성되는 순간 및/또는 개별 명성 표시자의 최신 업데이트 순간을 나타내는 타임스탬프(TS0로 표시됨)를 추가로 포함할 수 있다. 엔트리(17)는 개별 명성 표시자의 유효 기간을 나타내는 명성 라이프타임 표시자(reputation lifetime indicator, RL)를 추가로 포함할 수 있다. 명성 데이터에 대해 제한된 라이프타임을 특정함으로써, 일부 실시예들은 상기 데이터의 주기적인 갱신(refresh)을 효과적으로 강제하고, 따라서 개별 엔터티와의 잠재적인 감염 확산을 포함한다. 라이프타임 표시자는 실행가능한 엔터티들 사이에서 변화될 수 있고; 악성(malicious) 또는 양성(benign)으로 입증된 일부 엔터티들의 명성은 무한한 라이프타임을 가질 수 있다. 엔터티 핑거프린트 및 명성 표시자는 이하에서 더욱 상세히 설명하도록 한다.

[0016] 일부 실시예들은 엔터티의 현재 명성과 개별 엔터티의 과거 명성(historical reputation, HR)을 구별한다. 상기 현재 명성은 클라이언트 시스템에 현재 위치하거나 실행중인 엔터티의 명성을 나타낸다. 상기 과거 명성은 각각의 실행가능한 엔터티의 다른 인스턴스에 대해 이전에 컴퓨팅 되고 데이터베이스(16a 및/또는 16b)에 저장된 명성 표시자의 값을 나타내기 위해 본 명세서에서 사용된다. 과거 명성은 다른 클라이언트 시스템들로부터 집계된 그리고/또는 과거 다른 시간에 컴퓨팅 된 명성 데이터를 포함할 수 있다. 과거 명성은 인간 보안 분석가가 개별 엔터티에 대해 결정한 명성을 포함할 수 있다. 이러한 과거 명성은 결정 과정에서 자동으로 결정된 명성보다 큰 가중치를 가질 수 있는데, 이는 과거 명성이 자동으로 결정된 명성보다 정확할 가능성이 더 높기 때문이다.

[0017] 도 1 및 도 2에 도시된 예시적인 명성 관리 시스템은 계층적 방식으로 구성된다. 지연을 최소화하고 사용자 경험을 향상시키기 위해, 클라이언트 시스템(10a-e)들은 먼저, 로컬 명성 데이터베이스(16b)에서 명성 데이터를 검색(look up)할 수 있고, 그러고 나서 필요한 경우, 중앙 명성 데이터베이스(16a)로부터 그러한 데이터를 요청할 수 있다. 그러므로 일부 실시예들에서는, 로컬 데이터베이스(16b)는 중앙 데이터베이스(16a)의 로컬 캐시(cache)로 간주될 수 있다. 다수의 클라이언트 시스템(10a-e)들로부터의 명성 데이터를 집계함으로써, 중앙 명성 데이터베이스(16a)는 새로운 위협에 관한 정보를 신속하게 획득하고, 다른 클라이언트 시스템에 이것을 배포할 수 있다.

[0018] 도 2에 도시된 바와 같은 구성들은 명성 데이터를 핸들링하는 환경 특정 방식을 가능하게 할 수 있다. 일부 실시예들에서, 로컬 명성 데이터베이스(16b)는 개별적인 격리된 환경에 특정적으로 맞춰진 명성 표시자들을 저장한다. 그러한 일 예에서, 기업 인트라넷의 클라이언트 시스템(10d-e)들은 Microsoft Office®과 같은 널리 사용되는 소프트웨어 어플리케이션 *X*를 운용한다. 어플리케이션 *X*는 개별 클라이언트 시스템이 인터넷에 연결되는 한 멀웨어에 취약한 실행 모듈 *Y*를 로딩한다. 클라이언트 시스템(10d-e)들이 인터넷에 연결되어 있지 않을 때(예를 들어서, 환경(12)이 주위 방어 수단에 의하여 보호될 때), 어플리케이션 *X*는 인터넷 연결과 관련된 취약성을 더 이상 겪지 않는다. 따라서, 그러한 취약성에 대하여 어플리케이션 *X*를 모니터링하는 것은 시스템(10d-e)들에서 (즉, 격리된 환경(12) 내에서) 필요하지 않을 수 있고, 반면에 그러한 모니터링은 인터넷에 직접 연결된 시스템들에서 중요할 수 있다. 동등하게, 어플리케이션 *X*는 환경(12)의 외부와 비교하여, 환경(12) 내에서 높은 신뢰도를 가질 수 있다.

[0019] 환경 특수성(environment-specificity)의 다른 예에서, 기업은 격리된 환경(12) 밖에서 보통 마주치지 않는 사유 소프트웨어 어플리케이션 *X*를 사용한다. 따라서 어플리케이션 *X*에 관련된 명성 데이터는 다른 클라이언트 시스템들에 의해서는 사용되지 않을 것이다. 일부 실시예들에서, 그러한 명성 데이터는 환경 특정 명성 데이터베이스(16b)에서만 저장되고 중앙 명성 데이터베이스(16a)에서는 그러하지 않는다. 그러한 구성은 환경(12) 안에서 작동하는 클라이언트들에 대해서 뿐만 아니라, 격리된 환경(12) 밖에서 작동하는 클라이언트들에 대해서도 데이터베이스 색인(lookup)의 효율성을 증가시킬 수 있다.

[0020] 도 4a는 본 발명의 일부 실시예에 따른, 도 1 및 도 2의 클라이언트 시스템(10a-e)들과 같은 클라이언트 시스템(10)의 예시적 하드웨어 구성을 보여준다. 클라이언트 시스템(10)은, 특히 엔터프라이즈 서버와 같은 기업 컴퓨팅 장치, 또는 개인용 컴퓨터나 스마트폰과 같은 최종 사용자 장치(end-user device)를 나타낼 수 있다. 도 4a는 설명적 목적으로 컴퓨터 시스템을 보여준다. 모바일 전화기 또는 웨어러블과 같은 다른 클라이언트 시스템들은 다른 구성을 가질 수 있다. 클라이언트 시스템(10)은 프로세서(32), 메모리 유닛(34), 입력 장치(36)들 세트, 출력 장치(38)들 세트, 저장 장치(40)들 세트, 및 네트워크 어댑터(42)들 세트(이들은 모두 컨트롤러 허브(44)에 의하여 연결됨)를 포함한다.

[0021] 프로세서(32)는 신호 및/또는 데이터의 세트로 산술 및/또는 논리 연산을 실행하도록 구성된 물리적 장치(예컨대, 반도체 기판에 형성된 멀티-코어 집적 회로, 마이크로프로세서)를 포함한다. 일부 실시예들에서, 이러한 논리 연산들은 프로세서 명령(예를 들어, 머신 코드 또는 다른 유형의 소프트웨어)의 시퀀스 형태로 프로세서(32)에 전달된다. 메모리 유닛(34)은 명령들을 수행하는 도중에 프로세서(32)에 의해 액세스되거나 생성되는 데이

터/신호들을 저장하는 비-일시적 컴퓨터-판독 가능 매체(예컨대, RAM)를 포함할 수 있다. 입력 장치(36)는 사용자가 클라이언트 시스템(10)으로 데이터 및/또는 명령들을 도입할 수 있게 하는 개별 하드웨어 인터페이스 및/또는 어댑터를 포함하는, 특히 컴퓨터 키보드, 마우스, 및 마이크를 포함할 수 있다. 출력 장치(38)는 특히 디스플레이 스크린과 스피커는 물론, 시스템(10)이 사용자에게 데이터를 통신하게 할 수 있는 그래픽 카드와 같은 하드웨어 인터페이스/어댑터를 포함할 수 있다. 일부 실시예들에서, 입력 장치(36)와 출력 장치(38)는 터치-스크린 장치들의 경우와 같이, 하드웨어의 공통적인 부품을 공유할 수 있다. 저장 장치(40)는 소프트웨어 명령들 및/또는 데이터의 비휘발성 저장, 판독, 및 기록을 가능하게 하는 컴퓨터-판독 가능 매체를 포함한다. 예시적인 저장 장치(40)는 자기 디스크 및 광 디스크 및 플래시 메모리 장치들은 물론, CD 및/또는 DVD 디스크들 및 드라이브들과 같은 소거 가능 매체를 포함한다. 네트워크 어댑터(42)들 세트는 클라이언트 시스템(10)이 네트워크(20, 120) 및/또는 다른 장치들/컴퓨터 시스템들에 연결될 수 있게 한다. 컨트롤러 허브(44)는 복수의 시스템 버스, 주변 버스 및 칩셋 버스, 및/또는 도시된 하드웨어 장치들의 내부 통신을 가능하게 하는 모든 다른 회로를 일반적으로 나타낸다. 예를 들어서, 허브(44)는 특히 프로세서(32)를 메모리(34)에 연결하는 노스브리지(northbridge), 및/또는 프로세서(32)를 장치(36-38-40-42)들에 연결하는 사우스브리지(southbridge)를 포함할 수 있다.

[0022] 도 4b는 도 1의 중앙 명성 서버(14a) 또는 도 2의 로컬 명성 서버(14b)를 나타낼 수 있는 명성 서버(14)의 예시적 하드웨어 구성을 보여준다. 서버(14)는 서버 프로세서(132), 서버 메모리(134), 서버 저장 장치(140)들 세트, 및 네트워크 어댑터(142)들 세트(이들은 모두 서버 컨트롤러 허브(144)에 의하여 연결됨)를 포함한다. 장치(132, 134, 140, 및 142)들의 작동은 상술한 장치(32, 34, 40 및 42)들의 작동을 투영할 수 있다. 예를 들어서, 서버 프로세서(132)는 신호 및/또는 데이터의 세트로 산술 및/또는 논리 연산을 실행하도록 구성된 집적회로를 포함할 수 있다. 서버 메모리(134)는 연산을 실행하는 도중에 프로세서(132)에 의해 액세스되거나 생성되는 데이터/신호들을 저장하는 비-일시적 컴퓨터-판독 가능 매체(예컨대, RAM)를 포함할 수 있다. 네트워크 어댑터(142)들은 서버(14)가 네트워크(20, 120)들과 같은 컴퓨터 네트워크로 연결되도록 할 수 있다. 일부 실시예들에서, 명성 서버(14)는 이하에서 추가로 나타내는 바와 같이, 클라이언트 시스템 상에서 실행되는 소프트웨어 구성 요소로 구성된다.

[0023] 도 5는 본 발명의 일부 실시예에 따른, 클라이언트 시스템(10)에서 실행되는 소프트웨어 객체들의 예시적인 세트를 보여준다. 게스트 운영 시스템(OS)(46)은 클라이언트 시스템(10)의 하드웨어에 인터페이스를 제공하고 소프트웨어 어플리케이션(52a-c 및 54)들 세트를 위한 호스트로서 역할하는 소프트웨어를 포함한다. OS(46)는 특히, Windows®, MacOS®, Linux®, iOS®, 또는 Android™와 같은 임의의 널리 이용가능한 운영 시스템을 포함할 수 있다. 어플리케이션(52a-c)들은 특히 워드 프로세싱, 이미지 프로세싱, 데이터베이스, 브라우저, 및 전자통신 어플리케이션과 같은 임의의 사용자 어플리케이션을 일반적으로 나타낸다. 일부 실시예에서, 보안 어플리케이션(54)은 컴퓨터 보안 위협으로부터 클라이언트 시스템(10)을 보호하기 위하여 후술하는 바와 같이 안티-멀웨어 및/또는 다른 작업을 수행하도록 구성된다. 보안 어플리케이션(54)은 독립형 프로그램(standalone program)일 수 있고, 또는 소프트웨어 스위트(software suite)의 일부를 구성할 수 있다. 보안 어플리케이션(54)은 커널 레벨의 프로세서 권한에서, 적어도 부분적으로 실행될 수 있다.

[0024] 도 5에 도시된 것에 대하여 대안적인 실시예에서, OS(46) 및 어플리케이션(52a-c)들은 클라이언트 시스템(10) 상에서 실행되는 하이퍼바이저에 의해 노출된 가상 머신(VM) 내에서 실행될 수 있다. 그러한 실시예는 특히 서버 팜(server farm) 및 서비스로서의 인프라스트럭처(infrastructure as a service, IAAS)와 같은 클라우드 기반 아키텍처를 보호하는데 적합할 수 있다. 가상 머신은 물리적 컴퓨팅 시스템의 추상화(abstraction)(예를 들어, 소프트웨어 에뮬레이션(emulation))로 본 기술분야에서 일반적으로 알려져 있으며, 상기 VM은 가상 프로세서, 가상 저장소 등을 포함한다. 그러한 실시예에서, 보안 어플리케이션(54)은 개별 VM 내부에서 또는 외부에서 실행될 수 있다. 외부에서 실행되는 경우, 보안 어플리케이션(54)은 하이퍼바이저의 프로세서 권한 레벨에서 또는 별개의 가상 머신 내에서 실행될 수 있다. 단일 보안 어플리케이션은 개별 클라이언트 시스템 상에서 실행되는 복수의 VM들을 보호할 수 있다.

[0025] 도 6은 본 발명의 일부 실시예들에 따른 보안 어플리케이션(54)의 예시적인 구성 요소들을 보여준다. 어플리케이션(54)은 명성 관리자(58)에 통신가능하게 연결된 안티-멀웨어 엔진(56)을 포함한다. 안티-멀웨어 엔진(56)은 클라이언트 시스템(10)이 악성 소프트웨어를 포함하고 있는지를 결정하도록 구성된다. 일부 실시예들에서, 엔진(56)은 추가로 멀웨어를 제거하거나 그렇지 않으면 불능화시킬 수 있다. 멀웨어 탐지를 수행하기 위해, 엔진(56)은 본 기술분야에 알려져 있는 임의의 방법을 채용할 수 있다. 안티-멀웨어 방법은 일반적으로 두 가지의 넓은 카테고리에 속한다: 콘텐트 기반 및 행동적. 콘텐트 기반 방법은, 일반적으로 시그니쳐로 보통 알려져 있

는, 멀웨어를 나타내는 패턴에 대해 소프트웨어 엔터티의 코드를 스캔한다. 행동적 방법은 일반적으로 개별 엔터티에 의해 수행되는 특정 멀웨어를 나타내는 활동들을 탐지하기 위해 실행되는 엔터티를 모니터링한다. 소프트웨어 엔터티는, 이것이 악성 작용, 예를 들어 프라이버시의 상실, 개인적 데이터 또는 민감 데이터의 상실, 또는 사용자 입장에서 생산성의 저하를 조장하는 작동들의 세트 중 임의의 것을 수행하도록 구성된다면 악성으로 생각될 수 있다. 일부 예들에는 사용자에게 알리지 않거나 사용자의 승인없이 데이터를 수정, 삭제, 또는 암호화하는 것, 클라이언트 시스템(10)에서 실행되는 합법적 프로그램의 실행을 변경하는 것이 포함된다. 악성 작용의 다른 예들에는 특히 패스워드, 로그인 상세, 신용 카드나 은행 계좌 데이터, 또는 비밀 문서와 같은 사용자의 개인 정보나 민감한 데이터를 유출(extraction)하는 것을 포함한다. 악성 작동의 다른 예들은 제3자와 사용자의 대화 및/또는 데이터 교환에 대한 비인가 중간차단 또는 그 밖의 도청을 포함한다. 다른 예들은 요청하지 않은 통신(스팸, 광고)을 보내기 위하여 클라이언트 시스템(10)을 사용하는 것과 서비스 거부 공격(denial of service attack)에서와 같은 원격 컴퓨터 시스템에 악성 데이터 요청을 전송하기 위하여 전송하기 위하여 클라이언트 시스템(10)을 사용하는 것을 포함한다.

[0026] 일부 실시예들에서, 엔진(56)은 클라이언트 시스템(10) 상에 위치하는 그리고/또는 실행되는 실행가능한 엔터티들의 세트를 모니터링 및/또는 분석한다. 예시적인 실행가능한 엔터티들은 특히 어플리케이션, 프로세스, 및 실행 모듈을 포함한다. 실행 모듈은 프로세스의 구성 요소(component) 또는 빌딩 블록(building block)이며, 상기 개별적인 구성 요소는 실행 코드를 포함한다. 실행 모듈들은 개별 프로세스의 개시(launch) 및/또는 실행 중에 메모리로 로딩 및/또는 메모리로부터 언로딩 될 수 있다. 예시적인 실행 모듈은 특히 (Windows®의 EXE 파일과 같은) 프로세스의 메인 실행 파일(main executable), (DLL, dynamic-linked library와 같은) 공유 라이브러리를 포함한다. 일부 실시예에서, 프로세스의 메인 실행 모듈은 개별 프로세스가 개시될 때 실행되는 제1 기계 명령(machine instruction)을 포함한다. 라이브러리들은 프로그램의 여러 기능적 태양들을 실행하는 코드의 독립적인 섹션(self-contained section)들이다. 공유된 라이브러리들은 하나 이상의 프로그램에 의하여 독립적으로 사용될 수 있다. 실행가능한 엔터티들의 다른 예들은, 특히 개별 프로세스에 의해서 호출되는 실행 스크립트(예를 들어서, Perl, Visual Basic®, JavaScript® 및 Python 스크립트)와, 해석 파일(interpreted file)(예를 들어서, Java® JAR 파일들), 및 다른 엔터티들에 의해 개별 프로세스로 삽입된 코드의 조각을 포함한다. 코드 삽입(code injection, 코드 인젝션)은 개별 엔터티의 원래 기능을 변경하기 위해 다른 엔터티의 메모리 공간으로 코드의 시퀀스를 도입하는 방법군을 나타내기 위한, 본 기술 분야에서 사용되는 일반적인 용어이다. 본 기술 분야의 통상의 기술자라면 본 명세서에서 설명되는 시스템과 방법들이 다른 종류의 실행 모듈로 변환될 수 있다는 것을 알 수 있을 것이다.

[0027] 일부 실시예들에서, 명성 관리자(58)는 어플리케이션, 프로세스, 및 라이브러리를 포함하는 다양한 실행가능한 엔터티들(소프트웨어 객체)에 대한 명성 데이터를 결정하고, 그러한 데이터를 명성 데이터베이스에 저장 및/또는 명성 데이터베이스로부터 검색하고, 안티-멀웨어 엔진(56)으로 그러한 데이터를 전송하도록 구성된다. 일부 실시예들에서, 명성 관리자(58)는 엔터티 관리자(62), 활동 모니터(64), 핑거프린트 계산기(66), 및 명성 업데이트 스케줄러(68)를 포함한다. 이들 구성 요소들의 동작은 이하에서 더욱 상술될 것이다. 도 6에 도시된 것에 대하여 대안적인 실시예에서, 엔터티 관리자(62) 및 활동 모니터(64)는 안티-멀웨어 엔진(56)의 일부일 수 있다.

[0028] 일부 실시예들에서, 명성 관리자(58)에 통신 가능하게 연결된 클라이언트 명성 데이터베이스(16c)는 개별 클라이언트 시스템의 컴퓨터 판독가능 매체 상에 명성 데이터를 임시적으로 저장하도록 구성된다. 클라이언트 명성 서버(14c)는 클라이언트 시스템(10) 상에서 실행되는 컴퓨터 프로그램을 포함하고, 서버(14c)는 클라이언트 명성 데이터베이스(16c)에 명성 데이터를 선택적으로 추가 및/또는 검색하도록 구성된다. 데이터베이스(16c)는 전술한 데이터베이스 계층(hierarchy)의 일부를 형성하고, 적어도 부분적으로, 로컬 및/또는 중앙 명성 데이터베이스(16a-b)들의 캐시로서 기능할 수 있다. 도 6에 도시된 예시적인 구성에서, 명성 관리자(58)는 원격 서버(14a-b)들과 데이터를 교환하기 위해 통신 관리자(69)를 채용한다.

[0029] 도 7은 관리자(58)와 엔진(56) 사이의 예시적인 데이터 교환을 보여준다. 명성 관리자(58)는 안티-멀웨어 엔진(56)과 협력하여, 예를 들어, 타겟 엔터티와 연관된 명성 표시자(60)를 엔진(56)에 통신함으로써 안티-멀웨어 작동의 효율을 증가시킨다. 일부 실시예들에서, 명성 표시자(60)는 개별적인 실행가능한 엔터티가 악성일 확률을 나타낸다. 예시적인 명성 표시자(60)들은 최소값(예를 들어, 0)에서부터 최대값(예를 들어, 100)의 범위의 수치 명성 스코어(numerical reputation score)를 포함한다. 예시적인 일 실시예에서, 높은 명성 스코어는 개별 엔터티가 양성(악성 아님)일 가능성이 높음을 나타내는 반면, 낮은 스코어는 악성 의심 또는 악성일 가능성을 알 수 없음/현재 불확실함을 나타낸다. 다른 실시예들은 낮은 스코어가 높은 스코어보다 높은 신뢰도를 나타내

는 역스케일(reversed scale)을 사용할 수 있다. 명성 표시자는 최소치와 최대치 사이에서 연속적으로 변화할 수 있거나, 또는 소정의 개별 플래토(plateau)(예컨대, 10, 25, 50, 100)의 세트 사이에서 급변할 수 있다. 다른 실시예에서, 명성 표시자(60)는 복수의 라벨, 예컨대, "신뢰할 수 있음(신뢰)", "보통 신뢰할 수 있음", "신뢰할 수 없음(비신뢰)", 및 "알 수 없음"으로부터 값을 취할 수 있다.

[0030] 명성 표시자(60)를 수신하는 것에 응답하여, 안티-멀웨어 엔진(56)의 일부 실시예들은 비신뢰된 또는 알 수 없는 엔터티들과는 반대로 신뢰된 엔터티들을 우선적으로 처리한다. 예를 들어서, 엔진(56)은 신뢰된 객체를 스캔/모니터링하기 위하여 완화된 보안 프로토콜을 사용할 수 있고 알려지지 않았거나 또는 비신뢰된 객체를 스캔/모니터링하기 위하여서는 강화된 보안 프로토콜을 사용할 수 있고, 상기 완화된 보안 프로토콜은 상기 강화된 보안 프로토콜에 비하여 컴퓨팅 비용이 적다. 그러한 일 예에서, 완화된 보안 프로토콜은 엔진(56)이 신뢰된 객체를 스캔하기 위해서 멀웨어 탐지 방법들의 서브세트만 그리고/또는 멀웨어-식별 시행착오법의 서브세트만 채용하도록 명령할 수 있고, 반면에 강화된 보안 프로토콜은 엔진(56)에 이용가능한 방법들 및/또는 시행착오법의 모든 세트를 사용할 수 있다. 컴퓨팅 비용은 일반적으로 특정 절차를 실행하는 데 필요한 메모리 및/또는 프로세서 클럭 주기(processor clock cycle)의 수에 따라 표현(formulated)될 수 있다. 따라서 더 많은 클럭 주기 및/또는 더 많은 메모리를 요구하는 절차/프로토콜은 더 적은 클럭 주기 및/또는 더 적은 메모리를 요구하는 절차/프로토콜보다 컴퓨팅 비용이 더 높다고 여겨질 수 있을 것이다.

[0031] 일부 실시예들에서, 명성 표시자(60)는 예를 들어서 개별적인 실행가능한 엔터티에 의해 수행되는 다양한 활동에 응답하여 시간에 따라 변한다. 높은 명성이 신뢰를 나타내는 일 예에서, 개별적인 엔터티가 임의의 멀웨어를 나타내는 활동을 수행하지 않는다면, 타겟 엔터티의 명성은 시간에 따라 증가한다. 개별적인 명성은 또한 타겟 엔터티의 특정 활동에 대한 응답으로 감소할 수도 있다. 일부 실시예들에서, 타겟 엔터티의 명성은, 예를 들어, 개별 엔터티의 자식 엔터티에 의해 수행되는 멀웨어를 나타내는 활동에 대한 응답, 다른 엔터티로부터 코드 삽입을 받는 것에 대한 응답 등과 같은, 개별적인 타겟 엔터티와 관련된 다른 엔터티들의 활동에 대한 응답으로 바뀔 수 있다. 명성 관리자(58)는 도 7에 도시된 바와 같이, 안티-멀웨어 엔진(56)으로부터 타겟 엔터티의 다양한 활동에 관한 보안 통지를 수신할 수 있다.

[0032] 일부 실시예들에서, 명성 관리자(58)는 명성 데이터베이스의 계층에서 타겟 엔터티의 명성 표시자를 검색한다. 통신 지연 및 데이터 트래픽을 최소화하기 위하여, 명성 관리자(58)는 먼저 클라이언트 데이터베이스(16c)로부터 명성 데이터를 검색하려고 시도할 수 있다. 클라이언트 데이터베이스(16c)에서 매칭되는 데이터를 찾지 못한 경우, 그 후 관리자(58)는 로컬 데이터베이스(16b)를 쿼리(query)할 수 있다. 그러고 나서 찾고 있는(sought-after) 데이터가 여전히 발견되지 않는 경우, 관리자(58)는 원격, 중앙 명성 데이터베이스(16a)로부터 그것을 요청 진행할 수 있다. 도 8은 클라이언트 시스템(10)과 원격 명성 서버(14)(일반적으로 각각 도 1, 도 2 및 도 6의 서버들(14a-b-c)을 나타냄) 사이의 데이터 교환을 도시한다. 일부 실시예들에서, 클라이언트와 원격 명성 서버 사이의 이러한 통신은 중간자 공격(man-in-the-middle attack)을 피하기 위해 암호화 된다. 클라이언트 시스템(10)은 명성 요청(71)을 서버(14)에 전송할 수 있고, 요청(71)은 타겟 엔터티의 엔터티 핑거프린트와 같은 식별 토큰(identification token)을 나타낸다. 응답으로, 서버(14)는 데이터베이스(16)(일반적으로 각각 도 1 및 도 2의 데이터베이스(16a 및/또는 16b)를 나타냄)로부터 개별 타겟 엔터티에 대응하는 명성 표시자(60)를 선택적으로 검색할 수 있고, 클라이언트 시스템(10)에 표시자(60)를 전송할 수 있다. 클라이언트 시스템(10)은 또한 명성 리포트(73)를 서버(14)에 전송할 수 있고, 리포트(73)는 데이터베이스(16)에 저장하기 위하여 의도된 업데이트된 명성 표시자를 나타낸다.

[0033] 실행가능한 엔터티들과 명성 표시자들 사이의 명확한 연관성을 허용하기 위해, 각각의 실행가능한 엔터티는 본 명세서에서 엔터티 핑거프린트로 지칭되는 고유 토큰에 의한 방식으로 식별된다. 일부 실시예들에서, 핑거프린트 계산기(66)는 타겟 엔터티 및 실행 모듈에 대한 상기 핑거프린트를 계산하도록 구성된다. 핑거프린트는 본 기술분야에 알려져 있는 임의의 방법을 사용하여, 예를 들면, 해싱(hashing)을 통해, 생성될 수 있다. 해싱은 개별 객체의 해시로 알려져 있는 고정된 크기의 수 또는 비트 스트링을 얻기 위하여 객체의 일부분에 (예를 들어, 코드의 섹션 또는 전체 객체에) 해시 함수를 적용하는 것을 포함한다. 예시적인 해시 함수는 보안 해시(secure hash, SHA) 및 메시지 다이제스트 알고리즘(message digest(MD) algorithm)을 포함한다.

[0034] 바람직한 실시예에서, 엔터티 핑거프린트(70)는 개별 엔터티의 개별적인 구성 요소/빌딩 블록의 핑거프린트들의 세트에 따라 결정된다. 도 9에 도시된 바와 같은 예에서, 실행가능한 엔터티(80)는 실행 모듈(82a-c)들의 세트를 포함한다. 예를 들어서, Windows® 환경에서, 모듈(82a-c)들은 메인 실행 파일 및 두 개의 DLL을 각각 포함할 수 있다. 다른 예시적인 실시예에서, 모듈(82a-c)들은 다른 엔터티 구성 요소(예를 들어, 스크립트, JAR 파일, 삽입된 코드 조각 등)를 나타낼 수 있다. 본 기술분야의 통상의 기술자라면 본 명세서에서 설명되는 시스템

과 방법들이 다른 종류의 빌딩 블록과 다른 레벨의 단위(granularity)로 변환될 수 있다는 것을 알 수 있을 것이다.

[0035] 일부 실시예들에서, 모듈 핑거프린트(74a-c)(예를 들어, 해시)는 실행가능한 엔터티(80)의 구성 요소들 각각에 대해 계산된다. 그러고 나서 핑거프린트 계산기(66)는, 예를 들어 모듈 핑거프린트(74a-c)들을 정렬된 리스트로 배열함으로써 및/또는 모듈 핑거프린트(74a-c)들을 연쇄화(concatenating)함으로써, 모듈 핑거프린트(74a-c)들의 조합으로서 엔터티 핑거프린트(70)를 결정할 수 있다. 핑거프린트 비교 및 검색을 용이하게 하기 위해, 일부 실시예들은 모듈 핑거프린트(74a-c)들의 연쇄(concatenation)/리스트에 제2 해시 함수를 적용할 수 있다. 일부 실시예들에서, 엔터티 핑거프린트(70)는 경로 표시자의 리스트를 추가적으로 포함할 수 있고, 각각의 경로 표시자는 대응하는 구성 요소/모듈의 경로 또는 위치를 나타낸다. 개별 구성 요소가 삽입된 코드의 조각인 경우, 엔터티 핑거프린트(70)는 개별 조각의 메모리 주소 및/또는 크기를 인코딩할 수 있다.

[0036] 상기와 같이 구성된 각각의 엔터티 핑거프린트(70)는, 예를 들어 운영 시스템(46)에 의해 보여지는 것처럼 실행가능한 엔터티 그 자체보다는 구성 요소/빌딩 블록의 특정 구성(composition) 또는 배열을 특유하게 나타낸다. 통상적으로, 운영 시스템은 각각의 실행가능한 엔터티에 엔터티의 라이프타임 동안 개별 엔터티의 구성이 변경되는 경우에도 개별 엔터티의 전체 라이프타임 동안 불변인 채로 있는 고유 식별자(예를 들어, 프로세스 ID)를 할당한다. 대조적으로, 본 발명의 일부 실시예들에서는, 실행가능한 엔터티의 구성이 변경되는 경우(예를 들어, 프로세스가 라이브러리를 역동적으로 로딩하고 언로딩할 때), 엔터티 핑거프린트(70)와 그리고 개별 엔터티의 식별자(identity)가 그에 맞춰 변경될 수 있다. 달리 말하면, 일부 실시예들에서, 엔터티의 구성이 변경된 경우, 원래의 엔터티는 소멸되고, 새로운 엔터티가 생성된다. 일부 실시예들은 명성 표시자를 각각의 엔터티 핑거프린트와 고유하게 연관시키기 때문에, 실행가능한 엔터티의 구성이 변경되는 경우, 이들의 명성 또한 변경될 수 있다.

[0037] 도 10에 도시된 바와 같이, 구성 요소/빌딩 블록의 특정 조합이 다수의 실행가능한 엔터티에서 나타날 수 있다. 다른 엔터티 *X*의 모든 구성 요소를 가지는 엔터티 *Y*는 엔터티 *X*의 엔터티 초집합(superset)의 구성원이라고 본 명세서에서 언급된다. 도 9의 예에서, 세트(84a)는 엔터티(80a)의 엔터티 초집합인 반면, 세트(84b)는 엔터티(80a 및 80b)들 모두의 엔터티 초집합이다. 대조적으로, 엔터티(80d)는 모듈 A.exe.를 포함하지 않기 때문에, 엔터티(80d)는 각 엔터티(80a-c)들의 엔터티 초집합의 구성원이 아니다. 일부 실시예들에서, 엔터티의 명성은 개별 엔터티의 엔터티 초집합의 구성원들의 명성에 영향을 미칠 수 있으며, 결과적으로 이하에서 상세히 보여지는 바와 같이 상기 구성원들의 명성에 의해 영향을 받을 수 있다. 도 9의 예에서, 엔터티(80a)의 명성에서의 변화는 엔터티(80b-c)들의 명성에서의 변화를 일으킨다.

[0038] 일부 실시예들에서, 엔터티 관리자(62)(도 6)는, 클라이언트 시스템(10) 상에 위치하거나 그리고/또는 실행되는 복수의 실행가능한 엔터티들 뿐만 아니라 이러한 엔터티들 사이의 관계의 세트를 기술하고 본 명세서에서 명성 테이블(reputation table)로 불리는 데이터 구조를 유지한다. 예시적인 명성 테이블은 복수의 엔트리를 포함하고, 각각의 엔트리는 실행가능한 엔터티에 대응한다. 이러한 명성 테이블 엔트리(86) 중 하나가 도 11에 도시되어 있다. 엔트리(86)는 개별 엔터티의 엔터티 핑거프린트(70) 및 운영 시스템(46)에 의해 개별적인 실행가능한 엔터티에 할당된 엔터티 ID(entity ID, EID)를 포함한다. 개별적인 엔터티가 프로세스인 경우, 예시적인 EID는 Windows®에서 프로세스 ID(process ID, PID)를 포함한다. 이러한 구성은 핑거프린트(70)와 EID 사이의 즉각적인 연계를 허용하기 때문에 바람직할 수 있다. 엔터티의 구성은 시간에 따라 변경될 수 있기 때문에(예를 들어, 라이브러리를 역동적으로 로딩함으로써), 동일한 EID를 가지지만 고유한 핑거프린트를 가지는 다수의 명성 테이블 엔트리가 있을 수 있다. 또한, 클라이언트 시스템(10) 상에서 동시에 실행되는 동일한 엔터티의 다수의 인스턴스가 존재할 수 있으므로, 동일한 핑거프린트를 가지지만 고유한 EID를 가지는 다수의 명성 테이블 엔트리가 있을 수 있다. 원칙적으로, 각각의 이러한 객체는 그 자신의 행동 및 명성을 가질 수 있고, 그러므로 다른 객체와는 다르게 모니터링/분석될 수 있다.

[0039] 일부 실시예들에서, 엔트리(86)는, 예를 들어 개별 엔터티의 부모 엔터티의 식별자(부모 ID(parent ID, PID)) 및/또는 개별 엔터티의 자식 엔터티의 식별자와 같은 개별 엔터티의 계통 표시자(filiation indicator)를 추가적으로 저장할 수 있다. 예시적인 자식 엔터티는, 예를 들어 Windows® OS의 CreateProcess 함수를 통해 또는 Linux®에서 포크 메커니즘(fork mechanism)을 통해 부모 엔터티에 의해 생성된 자식 프로세스이다. 엔트리(68)는 또한 개별 엔터티 내로 코드를 삽입한 실행가능한 엔터티들의 식별자 세트 및/또는 개별 엔터티가 코드를 삽입한 엔터티들의 식별자 세트를 포함할 수 있다. 엔터티 핑거프린트일 수 있는 이러한 식별자들은, 삽입된 엔터티 ID(injected entity ID, INJID)로 표현된다.

[0040] 명성 테이블 엔트리(68)는 현재 엔터티의 엔터티 초집합의 구성원의 식별자 세트(초집합 구성원 ID(superset member ID, SMID))를 추가적으로 포함할 수 있다. 일부 실시예들에서, 각각의 SMID는 개별 초집합 구성원의 엔터티 핑거프린트로 구성될 수 있다. 선택적인 실시예에서, 각각의 SMID는 개별 엔터티 초집합 구성원과 관련된 명성 테이블 엔트리에 대한 포인터(pointer)를 포함할 수 있다. 핑거프린트(70)를 PID, SMID, 및/또는 INJID와 연관시키는 것은, 이하에서 더욱 상세히 보여지는 바와 같이 부모와 자식 엔터티 사이, 엔터티와 초집합 구성원 사이, 및 코드 삽입에 참여하는 엔터티들 사이에서 명성 정보의 전파를 용이하게 할 수 있다.

[0041] 타겟 엔터티의 현재 명성은 개별 엔터티의 행동에 따라, 그리고/또는 개별 엔터티의 다른 인스턴스의 행동에 따라 시간에 따라 상이할 수 있다. 일부 실시예들에서, 타겟 엔터티가 임의의 의심스러운 또는 멀웨어를 나타내는 활동을 수행하지 않는 경우, 개별 엔터티의 명성은 시간에 따라서, 예를 들면 소정의 스케줄에 따라서 증가할 수 있다. 명성 업데이트 스케줄러(68)(도 6)는, 예를 들어, 명성 표시자의 다음 업데이트가 일어나야 하는 순간을 결정하고, 현재 명성 표시자가 변경되어야 하는 증분(increment) ΔR을 결정함으로써 타겟 엔터티에 대한 명성 업데이트를 스케줄링하도록 구성될 수 있다.

[0042] 임시 데이터는 명성 테이블 엔트리(86)의 필드 세트에 (예를 들어, 타임스탬프로) 저장될 수 있다; 예컨대, 도 11의 시간 지시자(time indicator)(88) 참조. 이러한 시간 지시자 중 하나는 개별 엔터티 핑거프린트에 대응하는 명성 표시자의 최신 업데이트 시간을 나타낼 수 있다. 다른 시간 지시자는 개별 명성 표시자의 예정된 다음 업데이트에 대한 시간을 나타낼 수 있다. 따라서, 복수의 그러한 명성 업데이트 시간들은 각각의 타겟 엔터티의 명성 역학(reputation dynamics)을 상세히 시간순으로 기록할 수 있다. 또 다른 예시적인 시간 지시자는 개별 엔터티의 과거 명성의 만료 시간, 예를 들어 과거 명성에 대해 다음 데이터베이스 검색 시기가 된(due) 순간을 나타낼 수 있다. 과거 명성 라이프타임은 실행가능한 엔터티들 사이에서 다양할 수 있다. 캐시 명성 데이터에 대하여 제한된 라이프타임을 특정함으로써, 일부 실시예들은 로컬 또는 원격 명성 서버(14)로부터 명성 데이터의 갱신(refresh)을 효과적으로 강제하고, 따라서 잠재적인 감염을 포함한다.

[0043] 일부 실시예들에서, 활동 모니터(64)(도 6)는 클라이언트 시스템(10) 내에서 실행되는 어플리케이션 및 프로세스와 같은 엔터티들의 라이프-사이클 이벤트의 발생을 탐지하도록 구성된다. 예시적인 라이프-사이클 이벤트는 특히 실행가능한 엔터티의 개시 및/또는 종료, 개별 엔터티에 의한 라이브러리의 동적 로딩 및/또는 언로딩, 자식 엔터티의 스포닝(spawning, 생성), 및 코드 삽입을 포함한다.

[0044] 활동 모니터(64)는 객체간 관계들, 이를테면 어떤 프로세스가 어떤 실행 모듈을 로딩하는지, 어떤 엔터티가 어떤 엔터티의 부모 또는 자식인지, 어떤 엔터티가 어떤 엔터티로부터 삽입 코드를 받았는지 또는 삽입하였는지 등을 추가적으로 결정할 수 있다. 일부 실시예들에서, 활동 모니터(64)는 각 엔터티의 명성 테이블 엔트리(68)를 필요한 데이터(예를 들어, EID, PID, SMID, INJID 등)로 채우기(populate) 위하여 엔터티 관리자(62)와 협력한다. 엔터티 개시 탐지 및/또는 코드 삽입 탐지와 같은 과업을 수행하기 위하여, 모니터(64)는 특정 OS 함수를 호출하거나 후킹(hooking)하는 것과 같은 본 기술분야에서 알려진 임의의 방법을 채용할 수 있다. 예를 들어서, Windows® OS를 운영하는 시스템에서, 모니터(64)는 실행 모듈의 로딩을 탐지하기 위하여 LoadLibrary 함수로의 또는 CreateFileMapping 함수로의 호출을 중간차단(intercept, 인터셉트)할 수 있다. 다른 예에서, 모니터(64)는 새로운 프로세스의 개시를 탐지하기 위하여 PsSetCreateProcessNotifyRoutine 콜백을 등록하거나, 그리고/또는 삽입된 코드의 실행을 탐지하기 위하여 CreateRemoteThread 함수를 후킹할 수 있다.

[0045] 도 12a는 본 발명의 일부 실시예에서 명성 관리자(58)에 의해 수행되는 단계들의 예시적인 시퀀스를 보여준다. 단계들(302-304)의 시퀀스는 통지를 기다릴 수 있다. 일부 실시예들에서, 명성 관리자(58)는 프로세스 개시, DLL 로딩 등과 같은 엔터티 라이프-사이클 이벤트의 발생에 관하여 활동 모니터(64)에 의해 통지를 받는다. 관리자(58)는 또한 스케줄러(68)에 의해 특정 명성 테이블 엔트리가 업데이트 시기가 되었다(due)는 것을 통지받을 수 있다. 관리자(58)는 타겟 엔터티가 컴퓨터 보안에 관련될 수도 있는 특정 활동을 수행할 때, 안티-멀웨어 엔진(56)으로부터 추가적으로 통지를 수신할 수 있다(도 7 참조). 통지를 수신한 경우, 단계(304)는 개별 통지의 소스 및/또는 유형을 식별할 수 있고, 개별 통지를 일으킨 타겟 엔터티 및/또는 개별 통지에 의해 영향받게 될 엔터티를 추가로 식별할 수 있다. 일부 실시예들에서, 엔터티 모니터(64)는 현재 실행 중인 각각의 엔터티를 나타내기 위하여, OS(46)에 의해 사용되는 데이터 구조로부터 그러한 엔터티들의 신원을 결정할 수 있다. 예를 들어서, Windows에서는, 각각의 프로세서는 실행 프로세스 블록(executive process block, EPROCESS)으로서 표시되고, 이것은, 특히 개별 프로세스의 쓰레드들 각각에 대한 핸들링(handle), 및 OS(46)로 하여금 복수의 실행 프로세스들로부터 개별 프로세스를 식별할 수 있게 하는 고유 프로세스 ID를 포함한다. 유사한 프로세스 표시(representation)는 Linux® 및 다른 운영 시스템에서 이용가능하다. 하나 이상의 엔터티가 통지에 의하여 영향을 받을 때, 단계(304)는 개별 엔터티들 사이의 관계를 결정하는 것을 추가적으로 포함할 수 있다. 예를

들어서, 부모 프로세스가 자식 프로세스를 개시할 때, 엔터티 모니터(64)는 자식과 부모의 신원과, 이들의 관계(계통)의 유형을 기록할 수 있다.

[0046] 도 12b는 활동 모니터(64)로부터 통지를 수신하는 것에 응답하여 명성 관리자(58)에 의해 수행되는 단계들의 예시적인 시퀀스를 보여준다. 이러한 통지는 통상적으로 타겟 엔터티에 관한 라이프-사이클 이벤트의 발생을 통신한다. 단계(322)에서, 핑거프린트 계산기(66)는 개별 타겟 엔터티의 엔터티 핑거프린트를 계산할 수 있다. 단계(322)는 타겟 엔터티의 모듈/빌딩 블록을 리스팅하는 것, 각각의 그러한 모듈을 유지하는 메모리 섹션을 식별하는 것, 모듈 핑거프린트를 계산하는 것, 및 개별 모듈 핑거프린트에 따라서 엔터티 핑거프린트를 어셈블링하는 것을 포함할 수 있다(도 9 및 관련 설명 참조). 단계(323)에서, 엔터티 관리자(62)는 동일한 EID를 가지는 객체가 이미 추적/분석되고 있는지 여부를 결정하기 위하여, 명성 테이블에서 타겟 엔터티의 엔터티 ID(entity ID, EID)를 검색할 수 있다. 엔터티 ID는 타겟 엔터티를 식별하기 위하여 운영 시스템에 의해 사용된다. Windows® 환경에서, 예시적인 EID는 현재 실행 중인 프로세스의 프로세스 ID(process ID, PID)이다. 개별 EID가 새로운 경우(타겟 엔터티가 실행가능한 객체의 새로운 인스턴스임을 나타냄), 단계(325)에서, 엔터티 관리자(62)는 타겟 엔터티를 나타내기 위하여 새로운 명성 테이블 엔터티를 생성할 수 있다. 개별 EID가 새롭지 않은 경우(예를 들어, 프로세스가 라이브러리를 로딩하는 것과 같이 타겟 엔터티의 모듈 구성이 변경되는 경우), 단계(324)는 명성 테이블이 현재 타겟 엔터티와 동일한 핑거프린트(70)를 가지는 엔터티를 열거(list)하고 있는지 여부를 결정할 수 있다. 명성 테이블이 동일한 핑거프린트를 가지는 엔트리를 이미 포함하고 있는 경우, 명성 관리자(58)는 후술하는 단계(326)로 진행할 수 있다. 이러한 상황은, 예를 들어, 탐지된 라이프사이클 이벤트가 이미 실행중인 타겟 엔터티를 나타내는 경우에 발생할 수 있다. 타겟 엔터티의 핑거프린트가 새로운 경우(동일한 핑거프린트를 가지는 엔터티가 명성 테이블에 열거되지 않은 경우), 엔터티 관리자(62)는 개별 타겟 엔터티에 대하여 새로운 테이블 엔트리를 생성할 수 있다.

[0047] 일부 실시예에서, 엔터티의 모듈 구성에서의 변화는 엔터티 핑거프린트에서의 변화를 야기한다. 그러므로, 비록 개별 엔터티가 종료되지 않았어도 핑거프린트의 관점에서 보면 구 엔터티(old entity)는 마치 소멸된 것처럼 보이고, 새로운 엔터티가 클라이언트 시스템(10)에 출현한 것처럼 보일 수 있다. 이러한 경우에, 새로운 엔터티가 개시된 경우 뿐만 아니라, 단계(336)에서, 명성 관리자(58)가 개별 엔터티 핑거프린트와 관련된 과거 명성 데이터를 검색하려고 시도할 수 있다. 단계(336)는 예를 들어, 명성 관리자(58)가 명성 서버(14)에 명성 요청(71)을 보내는 것을 포함할 수 있다(예컨대, 도 8 참조). 개별 핑거프린트에 대하여 과거 명성 데이터가 존재하는 경우, 서버(14)는 데이터베이스(16)로부터 그러한 데이터를 선택적으로 검색하고, 표시자(60)를 클라이언트 시스템(10)에 전송할 수 있다. 이러한 상황은 개별 엔터티의 인스턴스(실행 모듈의 조합)가 어쩌면 특징적인(distinct) 클라이언트 시스템에서 실행되어 이전에 발견되었을 때, 그리고 개별 엔터티의 명성이 계산되고 데이터베이스(16)에 저장되었을 때 발생할 수 있다. 명성 표시자(60)를 수신하면, 단계(338)에서, 명성 관리자(58)는 타겟 엔터티의 현재 명성 표시자를 개별 엔터티의 과거 명성에 따라 결정된 값으로 설정할 수 있다. 예시적인 일 실시예에서, 현재 명성은 과거 명성과 동일하게 설정된다.

[0048] 단계(337)에서 타겟 엔터티에 대해 이용가능한 과거 명성이 없다고 결정하는 경우, 명성 관리자는 단계(339)로 진행한다. 이러한 상황은, 예를 들어 새로운 소프트웨어가 시장에 출현하는 경우(예컨대, 신제품 또는 소프트웨어 업데이트), 개별 엔터티에 대한 데이터베이스 엔트리가 만료된 경우, 또는 서버(14)를 이용할 수 없는 경우(예컨대, 네트워크 연결 부족, 서버 다운)에 발생할 수 있다. 단계(339)에서, 엔터티 관리자(64)는 타겟 엔터티가 명성 테이블에 현재 열거된 부모 엔터티의 자식 엔터티인지를 결정할 수 있다. "예"인 경우, 단계(340)에서, 일부 실시예들은 타겟 엔터티의 명성을 부모 엔터티의 명성에 따라 결정된 값으로 (예를 들어, 부모의 명성과 같거나 보다 낮게) 설정한다.

[0049] 단계(341)에서, 엔터티 관리자(64)는 명성 테이블에 현재 존재하는 타겟 엔터티의 엔터티 초집합의 임의의 구성원이 있는지 여부를 결정할 수 있다. "예"인 경우, 명성 관리자(58)의 일부 실시예들은 타겟 엔터티의 현재 명성을 초집합 구성원 엔터티의 명성에 따라 결정된 값으로 (예를 들어, 초집합 구성원의 명성과 같게) 설정한다. 그러한 명성 선택을 뒷받침하는 논리는 초집합 구성원들이 타겟 엔터티의 실행 모듈의 상당 부분 (또는 전체)을 포함하기 때문에, 타겟 엔터티의 명성이 초집합 구성원의 명성으로부터 추론될 수 있다고 간주하는 것이다.

[0050] 부모 엔터티 또는 초집합 구성원 엔터티가 존재하지 않는 경우, 단계(344)에서 명성 관리자(58)는 타겟 엔터티의 현재 명성을 소정의 디폴트 값(default value, 기본 값)으로 설정할 수 있다. 예를 들어, 알 수 없는 엔터티(unknown entity)의 명성은 낮은 신뢰도를 나타내는 값으로 설정될 수 있다(예컨대, 신뢰할 수 없음, 알 수 없음, R=0). 초기 명성은 또한 타겟 엔터티의 유형 또는 타겟 엔터티의 특징들의 세트에 의존할 수 있다. 예를 들면, 인터넷에서 다운로드된 엔터티는 디지털 서명이 되어있지 않으면 초기 명성 값 R=0을, 서명이 되어 있는 경

우에는 초기 명성 값 R=20%를 받을 수 있다.

[0051] 단계(326)에서, 업데이트 스케줄러(68)는 타겟 엔터티의 명성 테이블 엔트리의 다음 업데이트를 스케줄링 할 수 있다. 일부 실시예들에서, 타겟 엔터티의 명성은 시간에 따라 변할 수 있다. 예를 들어, 개별 엔터티가 의심스럽거나 또는 멀웨어를 나타내는 것으로 간주되는 임의의 활동을 수행하지 않는 경우, 및/또는 타겟 엔터티가 멀웨어를 나타내는 시그니처와 매칭되는 임의의 코드 패턴을 포함하지 않는 경우, 상기 개별 엔터티의 명성 표시자는 높은 신뢰 수준을 나타내는 값으로 진행할 수 있다(예컨대, R은 100% 신뢰 쪽으로 증가할 수 있음). 높은 R 값이 높은 신뢰를 나타내는 실시예에서의 명성 표시자에 대한 예시적인 변형 시나리오가 도 13에 도시된다. 도시된 명성 표시자는 소정의 값들 R_1, R_2, R_3 등의 세트 사이에서 급변할 수 있다. 명성에서의 그러한 변화는 예를 들어서, R이 시간 인스턴스 t_2에, 값 R_2에서 값 R_3로 증가할 수 있는 것과 같이(예컨대, 개별 타겟 엔터티의 생성 순간에 대해 측정됨), 소정의 순간에 발생할 수 있다.

[0052] 값 R은 개별 타겟 엔터티의 생성/개시 이후 경과된 시간에 따라 결정될 수 있다. 선택적인 실시예에서, R은 이전 이벤트(예컨대, 명성에서의 이전 증가, 보안 이벤트 등)의 발생 이후, 시간 간격 Δt가 경과한 후에 증가할 수 있다. 일부 실시예들에서, 시간 간격 Δt는 그 자체가 시간에 따라 변할 수 있다. 예를 들어, 명성 증가는 후기 단계보다 엔터티의 초기 라이프에서 덜 빈번할 수 있다. 다른 예에서, 시간 간격의 길이는 명성의 현재 값에 의존할 수 있다. 명성 증분(Reputation increments)은 현재 명성 값에 비례할 수 있다(예를 들어, 매번 R은 20% 증가할 수 있음). 명성 증분 ΔR은 또한 시간에 따라 다를 수 있다. 예를 들어서, R은 엔터티의 초기 라이프에서 소량 증가하고, 나중 시간에는 많은 양이 증가할 수 있다. 그러한 명성 역학을 뒷받침하는 근거는 악성 소프트웨어가 일반적으로 존재 초기 단계(즉, 개시 직후)에 이들의 활동을 수행하므로 엔터티가 충분히 오랜 시간 동안 잘 행동한다면 이것은 악성이 아니라고 가정하는 것이 안전할 수 있다는 것이다.

[0053] 일부 실시예들에서, 시간 간격 Δt 및/또는 명성 증분 ΔR은 개별 타겟 엔터티의 유형에 따라 변할 수 있다는 점에서, 엔터티-유형-특정적일 수 있다. 예를 들어서, 디지털 서명된 엔터티들의 명성 역학은 그렇지 않은 엔터티들의 명성 역학과는 다를 수 있다. 다른 예에서, 엔터티의 명성 역학은 개별 엔터티가 인터넷에 액세스하도록 구성되었는지 아닌지 여부에 따라 다를 수 있다.

[0054] 일부 실시예들에서, 명성 업데이트를 스케줄링하는 것은(도 12b의 단계(326)), 다음 업데이트 및/또는 명성 증가에 대한 시간 간격을 결정하는 것을 포함한다. 그러고 나서 단계(328)는 그에 맞춰 개별 엔터티의 명성 테이블 엔트리를 업데이트한다. 타겟 엔터티의 현재 명성에서의 변화는, 예를 들어 타겟 엔터티의 부모 엔터티 또는 타겟 엔터티의 초집합 구성원의 엔트리와 같은, 다른 엔터티의 현재 명성을 변경시키도록 유발할 수 있다. 그럴 경우, 단계(330)에서, 명성 관리자(58)는 그러한 업데이트를 수행한다. 단계들(332-334)의 시퀀스에서, 명성 관리자(58)는 명성 표시자(60)를 안티-멀웨어 엔진(56)으로 그리고 명성 서버(14)로 전송한다.

[0055] 도 12c는 업데이트 스케줄러(68)로부터의 통지(도 12a에 B로 표시됨)에 응답하여 명성 관리자(58)에 의해 실행되는 단계들의 예시적인 시퀀스를 보여준다. 이러한 통지는 통상적으로 타겟 엔터티를 식별하고, 개별 타겟 엔터티의 명성 표시자의 업데이트 시기가 되었다(due)는 것을 나타낸다. 단계(356)에서, 명성 관리자(58)는, 예를 들면, 개별 엔터티의 명성 테이블 엔트리의 필드에 저장된 명성 증분에 따라서, 개별 엔터티의 명성 표시자를 업데이트 할 수 있다(예컨대, 도 11 참조). 단계(358)에서, 명성 업데이트 스케줄러(68)는, 예를 들면, 시간 간격 Δt 및 명성 증분 ΔR을 결정하고, 이들 값을 개별 타겟 엔터티의 명성 테이블 엔트리의 대응 필드에 기록함으로써 다음 명성 업데이트를 스케줄링할 수 있다(단계 360). 명성 증분 ΔR은 절대값으로서 또는 현재 명성의 비율(예컨대, 20%)로서 결정될 수 있다. 단계들(360-364)의 시퀀스는 타겟 엔터티와 관련된 다른 엔터티들의 테이블 엔트리들을 업데이트하고, 명성 표시자(60)를 안티-멀웨어 엔진(56)에 전송한다.

[0056] 추가의 단계(366)에서, 명성 관리자(58)는 타겟 엔터티 및 가능하게는 다른 관련 엔터들의 명성의 변화를 반영하기 위해 명성 데이터베이스(16)의 업데이트를 촉발(trigger, 트리거)할 수 있다. 단계(366)는 업데이트된 명성 표시자를 포함하는 명성 리포트(73)를 명성 서버(14)에 전송하는 것을 포함할 수 있다(예를 들어, 도 8). 이러한 업데이트는 새로운 명성을 다른 인스턴스를 구동하는 다른 클라이언트 시스템에서 동일한 타겟 엔터티에 대해 사용할 수 있게 만들고, 그에 따라 클라이언트의 네트워크를 통해 컴퓨터 보안 지식을 전파한다. 명성 서버(14)가 리포트(73)를 다루는 예시적인 방식에 대해서는, 도 15와 관련하여 이하를 참조한다.

[0057] 도 12d는 안티-멀웨어 엔진(56)으로부터의 보안 통지에 응답하여 명성 관리자(58)에 의해 수행되는 단계들의 예시적인 시퀀스를 보여준다(예컨대, 도 7 참조). 이러한 통지는 안티-멀웨어 엔진이 특정 타겟 엔터티가 악성으로 의심된다고 결정할 때 생성될 수 있다. 일부 실시예들에서, 엔진(56)은 보안에 관련된 이벤트의 발생 또는

멀웨어를 나타내는 이벤트의 발생에 대하여 명성 관리자(58)에게 통지할 수 있다. 예시적인 이벤트는, 특히 메모리 액세스 허가를 위반하는 방식으로 메모리에 액세스하려는 시도, 운영 시스템의 특정 기능을 실행하려는 시도(예를 들어, 디스크 파일 생성, 레지스트리 엔트리 편집 등), 특정 작업을 수행하려는 시도(예를 들어, 다른 엔터티로의 코드 삽입, 원격 서버로부터의 파일 다운로드)를 포함한다. 통지(72)는 개별 이벤트에 의해 유발되거나 또는 영향받는 엔터티의 식별자 및개별 이벤트의 유형의 표시자를 포함할 수 있다. 통지의 다른 예는 타겟 엔터티의 코드를 파싱하는 동안 악성 코드 시그니쳐를 찾는 시그니쳐 스캐너에 응답하여 생성될 수 있다.

[0058] 보안 통지(72)를 수신하는 것에 응답하여, 단계(372)에서 명성 관리자(58)는 개별 타겟 엔터티의 명성 표시자에 대한 새로운 값을 결정할 수 있다. 일부 실시예들에서, 엔터티가 악성을 나타내거나 그렇지 않으면 개별 엔터티가 악성으로 의심되게 하는 활동을 수행하는 경우, 개별 엔터티의 명성은 낮은 신뢰도의 방향으로 변한다. 이러한 태양은 도 13에 도시되며, 여기서 R의 값은 보안 이벤트에 응답하여 감소(drop)한다. 감소의 크기는 규칙/보안 정책의 세트에 따라서 명성 관리자(58)에 의해 결정될 수 있다. 감소의 크기는 절대값으로서 또는 현재 명성 값의 비율(예컨대, 50%)로서 표현될 수 있다.

[0059] 일부 실시예들에서, 그러한 경우에 발생하는 명성에서의 감소의 크기는 이벤트의 유형 또는 보안 통지의 유형에 따라 다르다. 일부 이벤트/활동들은 보다 명백하게 멀웨어를 나타내고 따라서 명성에서 큰 감소를 촉발할 수 있다. 다른 이벤트들이 반드시 악성을 나타내지는 않아도, 다른 이벤트들과 함께 또는 타겟 엔터티에 의해 수행되는 특정 활동들과 함께 발생하는 경우, 그럴 수 있다. 그러한 이벤트 또는 활동에 의해 촉발된 명성의 변화는 명백한 악성 이벤트/활동과 관련된 것보다 상대적으로 작을 수 있다. 일부 보안 통지는 개별 타겟 엔터티에 대한 명성의 전체 손실(total loss)을 유발할 수 있다. 일부 실시예들에서, 명성에서 감소는 개별 명성 표시자가 과거에 다른 감소를 겪었는지에 따라, 이전 명성 감소 이후 경과된 시간에 따라, 및/또는 이전 명성 감소를 촉발한 보안 통지의 유형에 따라 결정될 수 있다. 일부 멀웨어 에이전트는 복수의 엔터티에 걸쳐 악성 활동을 조정하고 탐지를 피하기 위하여 그러한 활동을 제때 분산시킨다. 보안 통지의 이전 히스토리에 대해 현재 명성 감소를 조절하는 것은 일부 이러한 정교한 멀웨어 시나리오를 처리할 수 있다. 일부 실시예들에서, 단계(372)에서 발생하는 명성에서 변화는 타겟 엔터티의 현재 명성에 따라 및/또는 다른 엔터티의 현재 명성에 따라 계산된다. 그러한 일 예에서, 엔터티 *X*가 엔터티 *Y*로 코드를 삽입할 때, 두 엔터티 중 더 신뢰할 수 있는 것의 명성은 덜 신뢰할 수 있는 것의 현재 명성과 동일해질 수 있다.

[0060] 단계(374)에서, 명성 관리자(58)는 예를 들어, 시간 간격 Δt 및 명성 증분 ΔR을 생성함으로써 개별 타겟 엔터티의 명성의 업데이트를 스케줄링 할 수 있다. 추가의 단계(376)은 개별 엔터티의 명성 테이블 엔트리에 이러한 데이터를 저장할 수 있다. 일부 실시예들에서, Δt 및/또는 ΔR의 값은 보안 통지의 유형에 따라 다를 수 있다. 이러한 일 예에서, 엔터티가 명백히 악성을 나타내는 활동을 수행했을 때, 이것은 비교적 오랜 기간 동안 신뢰할 수 없는 것으로 머물게 된다. 대조적으로, 덜한 보안 필수 이벤트(less security-critical event)에 의해 유발된 감소 이후에는, 타겟 엔터티의 명성이 다시 비교적 빠르게 증가할 수 있다.

[0061] 일부 실시예들에서, 단계들(376-380-382)의 시퀀스는 타겟 엔터티(존재하는 경우)와 연관된 다른 엔터티의 명성 테이블 엔트리를 업데이트 할 수 있고, 명성 표시자(60)를 안티-멀웨어 엔진(56)에 전송할 수 있으며, 명성에서의 변화를 서버(14)에 보고할 수 있다.

[0062] 도 14는 본 발명의 일부 실시예에 따른 안티-멀웨어 엔진(56)에 의해 수행되는 단계들의 예시적인 시퀀스를 보여준다. 엔진(56)은 엔터티-특정 명성에 따라 멀웨어 탐지 및 예방, 및/또는 클린업(cleanup) 활동을 수행하도록 구성될 수 있다(단계 392). 달리 말하면, 안티-멀웨어 엔진(56)은 엔터티-특정 프로토콜/정책에 따라 각각의 실행가능한 엔터티를 모니터링 및/또는 분석할 수 있고, 여기서 개별 정책/프로토콜은 각각의 엔터티의 명성 표시자에 따라 하나의 엔터티에서 또 다른 엔터티까지, 다양할 수 있다. 일부 실시예들에서, 높은 신뢰도를 나타내는 명성을 가지는 엔터티는 낮은 신뢰도를 가지는 엔터티보다 낮은 컴퓨팅 비용 과정을 이용하여 분석될 수 있다.

[0063] 행동 멀웨어 탐지(Behavioral malware detection)는 통상적으로 타겟 엔터티가 악성인지 여부를 결정하기 위해 규칙들의 세트를 사용한다. 이러한 규칙은 종종 휴리스틱(heuristic)이라고도 한다. 하나의 예시적인 휴리스틱은, 예를 들어, 제1 엔터티가 제2 엔터티로 코드 조각을 삽입하고, 개별 코드가 인터넷으로부터 파일을 다운로드하려고 시도한다면, 상기 제1 엔터티는 아마도 악성이라고 말할 수 있을 것이다. 그러한 휴리스틱을 구현하기 위해, 안티-멀웨어 엔진(56)은 다양한 이벤트(예를 들어, 상기 예시에서 코드 삽입 및 원격 서버에 연결하려는 시도)들을 모니터링하는 것이 필요할 수 있다. 일부 이러한 이벤트는 다른 것보다 모니터링 하기 위한 컴퓨팅 비용이 크다. 또한, 일부 휴리스틱은 다른 것들보다 본질적으로 더욱 복잡하거나 그리고/또는 적용하기 더욱 어

려울 수 있다. 복잡한 휴리스틱은 예를 들어 "방법 A를 적용; A의 결과가 *X*이면, 방법 B를 적용; B의 결과가 *Y*이면, 추가로 조건 *Z* 등을 확인"과 같은 간단한 휴리스틱들의 조합을 포함할 수 있다.

[0064] 비용이 많이 드는 휴리스틱의 일부 예시로는, 랜섬웨어(ransomware)를 탐지하기 위해 사용되는 휴리스틱(모든 파일 시스템 활동 - 모든 파일 읽기, 쓰기, 및/또는 복사를 모니터링하는 것을 포함함) 및 OS 레지스트리 키에 관한 휴리스틱(예를 들어, 레지스트리에 대한 모든 쓰기를 인터셉트하는 것 및 특정 키를 변경하려는 시도를 포함하는지 여부를 결정하는 것을 포함함)을 포함한다. 비용이 많이 드는 휴리스틱의 또 다른 예시는, 자주 사용되는 OS 기능(예를 들어, CreateFile, ReadFile)에 대한 호출을 탐지하는 것이 필요하며, 이러한 호출을 탐지하는 것은 상당한 오버헤드를 초래할 수 있다. 대조적으로, 정기적인 작업(예를 들어, CreateRemoteThread)에서 아주 드물게 사용되는 OS 기능에 대한 호출을 탐지하는 것은 클라이언트 시스템(10)에 훨씬 낮은 부담을 줄 수 있다.

[0065] 일부 실시예들에서, 명성-의존 탐지 프로토콜을 얻는 것은, 명성 표시자에 따라서 이벤트 모니터링 및/또는 휴리스틱의 복잡성을 변화시키는 것을 포함한다. 달리 말하면, 안티-멀웨어 엔진(56)은 신뢰할 수 없는 엔터티 보다 더 적고 비교적 간단한 휴리스틱을 사용하여 신뢰할 수 있는 엔터티를 모니터링할 수 있다. 엔진(56)은 또한 신뢰할 수 있는 엔터티들을 모니터링 할 때, 특정 이벤트 또는 행동의 탐지를 비활성화시킬 수 있다. 콘텐트 기반 안티-멀웨어 방법은 예를 들어, 명성에 따라서 시그니쳐 데이터베이스의 크기를 조정함으로써, 명성-특정적으로 만들어질 수 있다. 그러한 일 예에서, 신뢰할 수 있는 엔터티들은 멀웨어를 나타내는 시그니쳐들의 비교적 작은 세트의 존재에 대해 검사될 수 있고, 반면 신뢰할 수 없는 엔터티들은 실질적으로 더 큰 시그니쳐 세트를 사용하여 검사될 수 있다.

[0066] 명성 표시자로 모니터링 프로토콜을 조정하는 일 예가 표 1에 도시된다.

[0067] 표 1

명성 표시자	프로토콜
0% 신뢰할 수 있음	최대 모니터링, 사용가능한 모든 휴리스틱 채용
10% 신뢰할 수 있음	일부 비용이 많이 드는 휴리스틱 비활성화
…	
80% 신뢰할 수 있음	코드 삽입 및 파일 드랍/복사 모니터링
90% 신뢰할 수 있음	코드 삽입에 대해서만 모니터링
100% 신뢰할 수 있음	전혀 모니터링 하지 않음

[0068]

[0069] 도 14로 돌아가서, 단계들(392-394)의 시퀀스에서, 안티-멀웨어 엔진(56)은 명성-특정적 프로토콜에서 기술된 바와 같이 이벤트의 발생을 대기하도록 구성된다. 그러한 보안-관련 이벤트 외에도, 엔진(56)은 명성 관리자(58)로부터 명성 표시자를 수신할 수 있다. 명성 표시자를 수신하는 것은 특정 엔터티의 명성이 변경되었음을 나타낼 수 있다. 명성 표시자를 수신하는 것에 응답하여(단계 396), 단계(398)에서 안티-멀웨어 엔진은 개별 타겟 엔터티를 식별하고, 수신된 명성 표시자의 값에 따라 개별 엔터티에 적용되는 모니터링 프로토콜/정책을 업데이트할 수 있다.

[0070] 탐지된 이벤트가 보안 이벤트(예를 들어, 엔터티가 다른 엔터티에 코드를 삽입한 경우)를 포함하는 경우, 단계(402)에서 안티-멀웨어 엔진(56)은 개별 이벤트를 유발한 타겟 엔터티 및/또는 개별 이벤트에 의해 영향받는 타겟 엔터티를 식별할 수 있다. 추가의 단계(404)는 타겟 엔터티의 신원 및 탐지된 이벤트의 유형에 따라 보안 통지를 만들고(formulate), 개별 보안 통지를 명성 관리자(58)에게 전송할 수 있다.

[0071] 도 15는 본 발명의 일부 실시예에 따른 명성 서버(14)(예컨대, 도 1 및 도 2의 서버들(14a-b))에 의해 수행되는 단계들의 예시적인 시퀀스를 보여준다. 단계들(412-414)의 시퀀스에서, 서버(14)는 클라이언트 시스템(10)으로

부터의 통신에 주목할 수 있다. 통신이 수신되는 경우, 단계(416)는 개별 통신이 명성 요청인지를 결정할 수 있다(예컨대, 도 8 참조). "예"인 경우, 서버(14)는 개별 요청에 포함된 엔터티 핑거프린트와 관련된 과거 명성 데이터를 검색하고, 요청 클라이언트에게 상기 데이터를 전송할 수 있다(단계들 418-420).

[0072] 통신이 명성 리포트를 포함하는 경우, 단계(424)에서, 서버(14)는 개별 명성 리포트에 포함된 엔터티 핑거프린트와 관련된 명성 데이터를 검색할 수 있다. 리포트(73)가 현재 명성 값이 데이터베이스(16)에 저장된 과거 명성 보다 낮은 신뢰를 나타내는 경우, 단계(428)에서 명성 서버(14)의 일부 실시예들은 클라이언트(10)로부터의 리포트에 수신한 명성 표시자의 값을 포함하도록 개별 데이터베이스 엔트리를 즉시 변경할 수 있다.

[0073] 리포트(73)가 현재 저장된 값보다 높은 신뢰를 나타내는 명성 표시자를 포함하는 경우, 일부 실시예들에서 단계(430)는 다양한 클라이언트들로부터 수신한 리포트들의 컬렉션에 명성 리포트(73)를 추가할 수 있다. 단계(432)에서, 명성 서버(14)는 이어서 업데이트 조건이 만족되는지를 결정하고, 상기 업데이트 조건이 만족되는 경우에만 데이터베이스 엔트리를 업데이트할 수 있다. 업데이트 조건은 시간 제약에 따라 및/또는 각각의 개별적인 엔터티 핑거프린트에 대해 수신된 리포트의 수에 따라 만들어질 수 있다. 예를 들어, 업데이트는 개별 엔터티 핑거프린트에 대응하는 명성 표시자의 최신 업데이트 이후 특정 시간 간격이 경과한 후에만 일어날 수 있다. 다른 예에서, 업데이트는 개별 타겟 엔터티에 관한 최신 보안 통지 이후 특정 시간 간격이 경과한 후에만 일어날 수 있다. 높은 명성이 높은 신뢰와 동등시되는 예시적인 일 실시예에서, 업데이트 조건이 만족되는 경우, 타겟 엔터티의 과거 명성은 최신 업데이트 기간 동안 개별 타겟 엔터티에 대해 보고된 모든 명성의 최소값과 동일한 값으로 업데이트 된다.

[0074] 위에서 상술한 예시적인 시스템과 방법들은 악성 소프트웨어로부터, 개인용 컴퓨터, 태블릿, 또는 스마트폰과 같은 클라이언트 시스템을 보호할 수 있게 한다. 일부 실시예들에서, 명성 관리자는 안티-멀웨어 엔진과 동시에 실행된다. 안티-멀웨어 엔진은 개별 클라이언트 시스템에서 실행되는 멀웨어 탐지 및/또는 그러한 멀웨어를 불능화(incapacitating)하거나 제거하는 것과 같은 작동을 수행한다. 클라이언트 시스템에서 실행되는 각각의 엔터티(예컨대, 어플리케이션, 프로세스, 스크립트)에 대하여, 명성 관리자는 안티-멀웨어 엔진에 명성 표시자를 전송할 수 있고, 상기 명성 표시자는 개별 엔터티가 악성이 아니라는 신뢰의 레벨을 표시한다.

[0075] 종래의 보안 시스템들에서, 소프트웨어 엔터티들은 그들의 명성에 관계없이 스캔 및/또는 모니터링된다. 반대로, 본 발명의 일부 실시예들에서, 안티-멀웨어 엔진은 신뢰된 엔터티에는 특혜적 처리를 해줄 수 있다. 예를 들어서, 안티-멀웨어 엔진은 신뢰되지 않은 또는 알 수 없는/이전에 본적 없는 엔터티와 비교하여, 신뢰된 엔터티를 스캔/모니터링 하는데 컴퓨팅 비용이 많이 드는 프로토콜(예를 들어, 더 많은 프로세서 클럭 사이클 및/또는 더 많은 메모리를 필요로 함)을 적게 사용할 수 있다. 그러한 일예에서, 규칙들의 서브세트(subset)는 신뢰된 엔터티들을 스캔/모니터링할 때 비활성화될 수 있다. 그러한 접근은 신뢰된 엔터티를 스캔/모니터링하는 것과 연계된 연산 부담을 경감함으로써 안티-멀웨어 성능을 실질적으로 개선할 수 있다.

[0076] 본 발명의 일부 실시예들에서, 각각의 실행가능한 엔터티는 구성 요소/빌딩 블록의 고유한 조합으로 보여진다. 그러한 빌딩 블록의 예에는, 특히 메인 실행 파일(main executable), 공유 라이브러리, 스크립트, 및 삽입된 코드 섹션이 포함된다. 구성 요소의 각 조합은, 예를 들어 개별 구성 요소들의 해시 조합을 포함하는 엔터티 핑거프린트를 통해 식별될 수 있다. 명성 표시자는 그러고 나서 각각의 엔터티 핑거프린트와 연관될 수 있다. 엔터티의 구성이 변경되는 경우(예를 들어, 프로세스가 라이브러리를 역동적으로 로딩하거나 삽입된 코드 조각을 수신하는 경우), 이들의 핑거프린트는 변경되고, 이들의 명성도 변경된다.

[0077] 일부 실시예들에서, 엔터티의 명성은 시간에 따라 변한다. 엔터티가 임의의 의심스럽거나 멀웨어를 나타내는 활동을 수행하지 않는 동안, 이들의 명성은 높은 신뢰를 나타내는 값으로 변경될 수 있다. 대조적으로, 엔터티가 멀웨어를 나타내거나 그렇지 않으면 보안 관련 활동을 수행하는 경우, 이들의 명성은 낮은 신뢰를 나타내는 값으로 격하될 수 있다. 그러한 명성에서의 변화는 로컬 캐시에 저장될 수 있고 그리고/또는 중앙 명성 데이터베이스로 전송될 수 있다. 그러한 구성은 명성의 임의의 모든 변화가 개별적 공유된 라이브러리의 인스턴스를 이용하여 다른 로컬 프로세스들로, 그리고 또한 명성 서버에 연결된 다른 클라이언트 시스템들에 추가로 빠르게 전파되도록 할 수 있다.

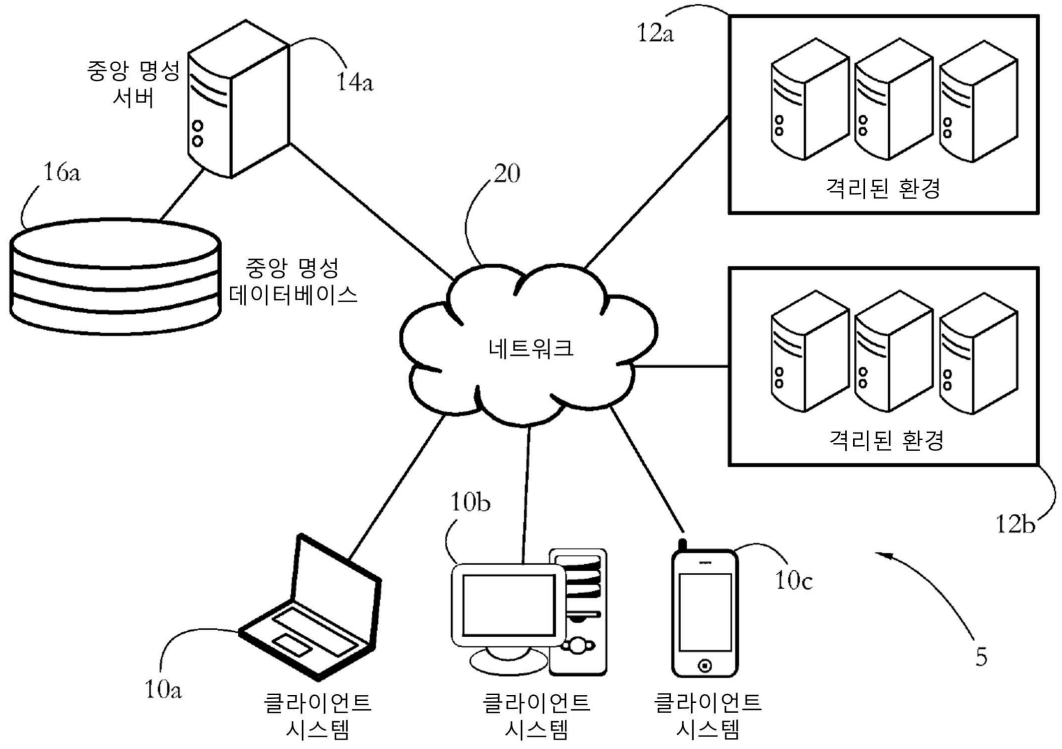
[0078] 일부 실시예들에서, 명성의 하락(악의가 의심됨을 나타낼 수 있음)은 명성 데이터베이스에, 그리고 이들로부터 다른 클라이언트 시스템으로 비교적 신속하게 전파되는 반면, 명성의 증가(신뢰의 증가를 나타낼 수 있음)는 보안 사고 없이 충분한 시간이 경과한 후에, 또는 충분한 수의 클라이언트 시스템에 의해 개별 엔터티가 행실이 바른것으로(well-behaved) 보고된 후에만 효력이 발생할 수 있다.

[0079] 본 명세서에서 설명된 시스템들과 방법들은 신흥 위협을 포함하는 광범위하고 다양한 악성 소프트웨어에 용이하게 적용될 수 있다. 또한, 명성 관리자는 안티-멀웨어 엔진과 독립적으로 작동하기 때문에, 안티-멀웨어 엔진은 명성 관리자의 작동에 영향을 주지 않으면서, 새로운 스캔/모니터링 방법 및 절차를 통합하도록 업그레이드 될 수 있다.

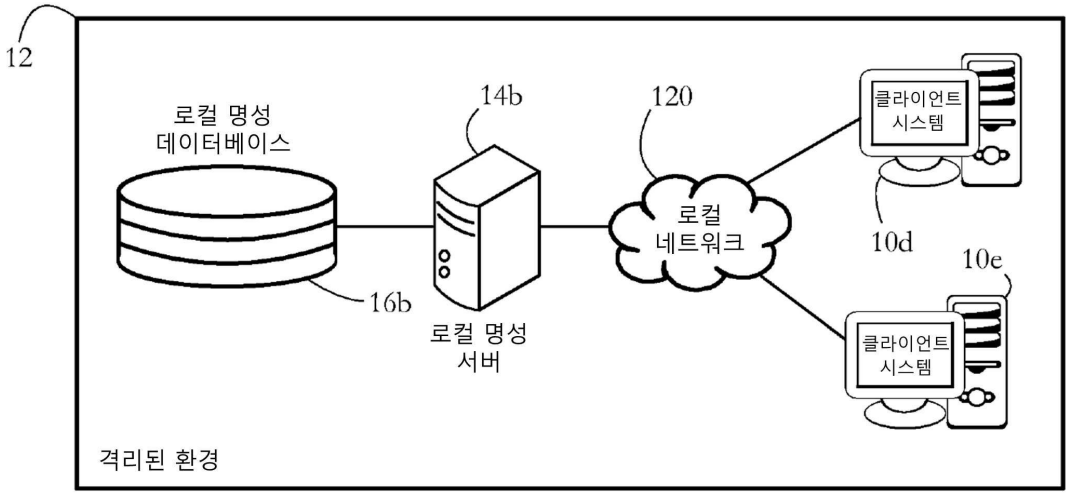
[0080] 본 발명의 범위를 벗어나지 않는 한 상기 실시예들은 다수의 방법으로 변경될 수 있음은 본 기술분야의 통상의 기술자에게 자명할 것이다. 따라서 본 발명의 범위는 이하의 청구항과 이들의 법적 균등물에 의하여 결정되어야 한다.

도면

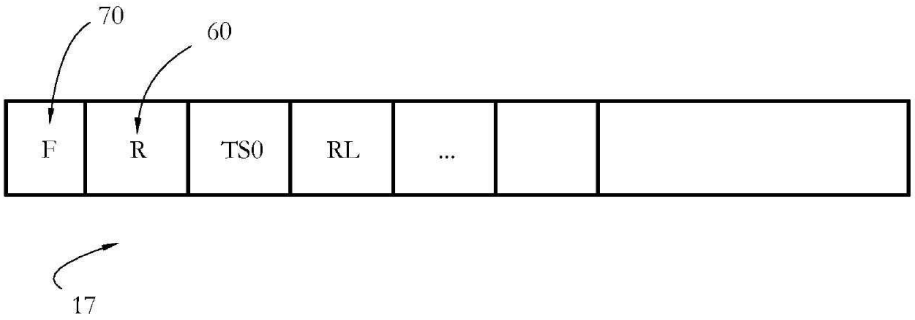
도면1



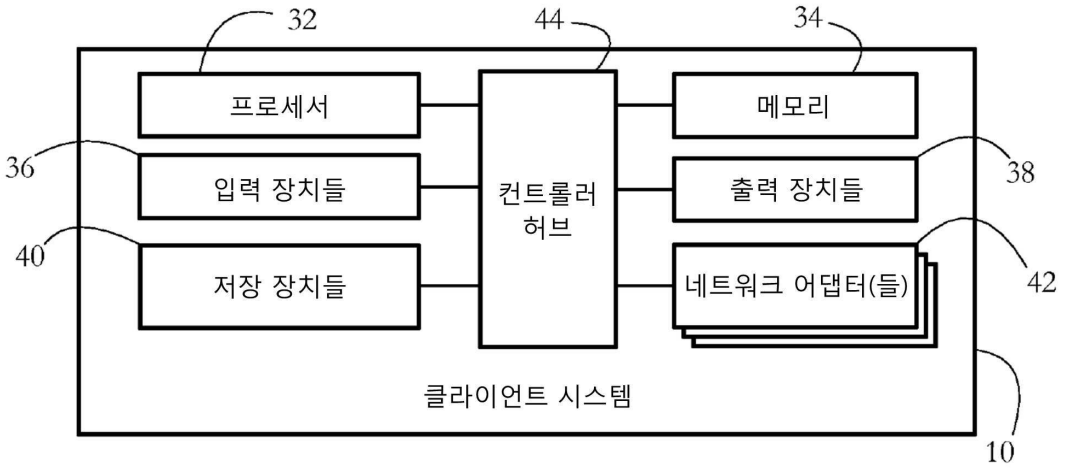
도면2



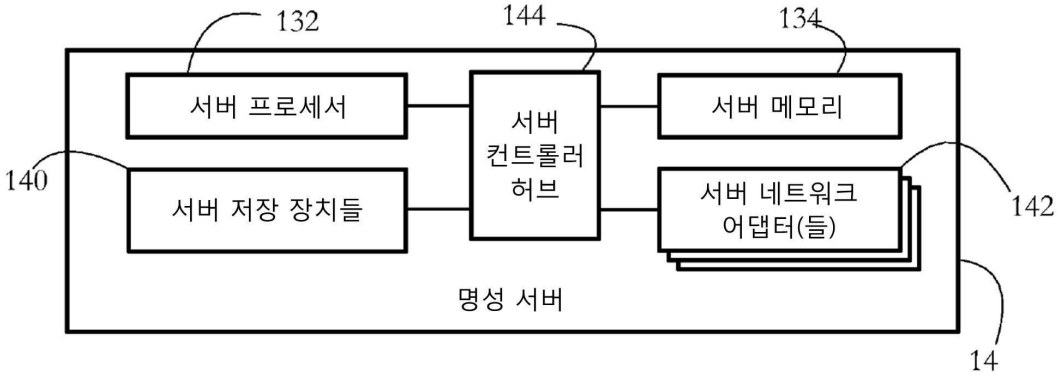
도면3



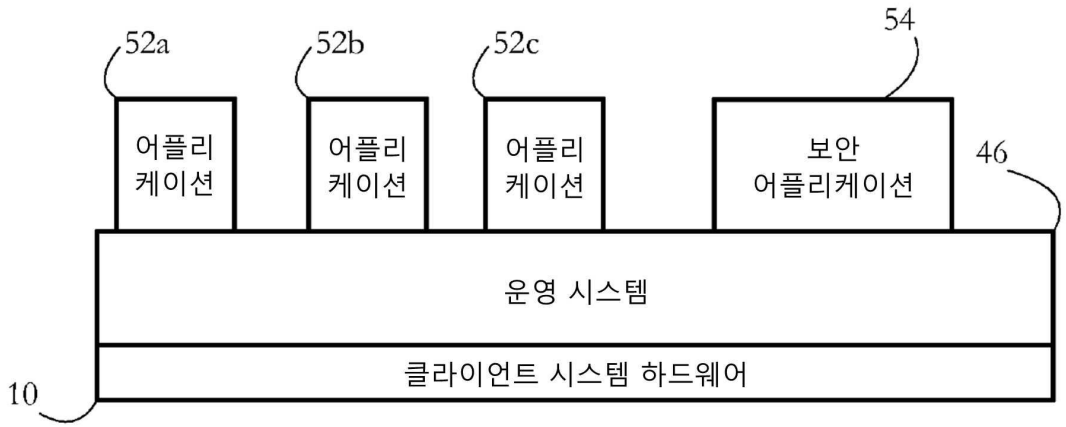
도면4a



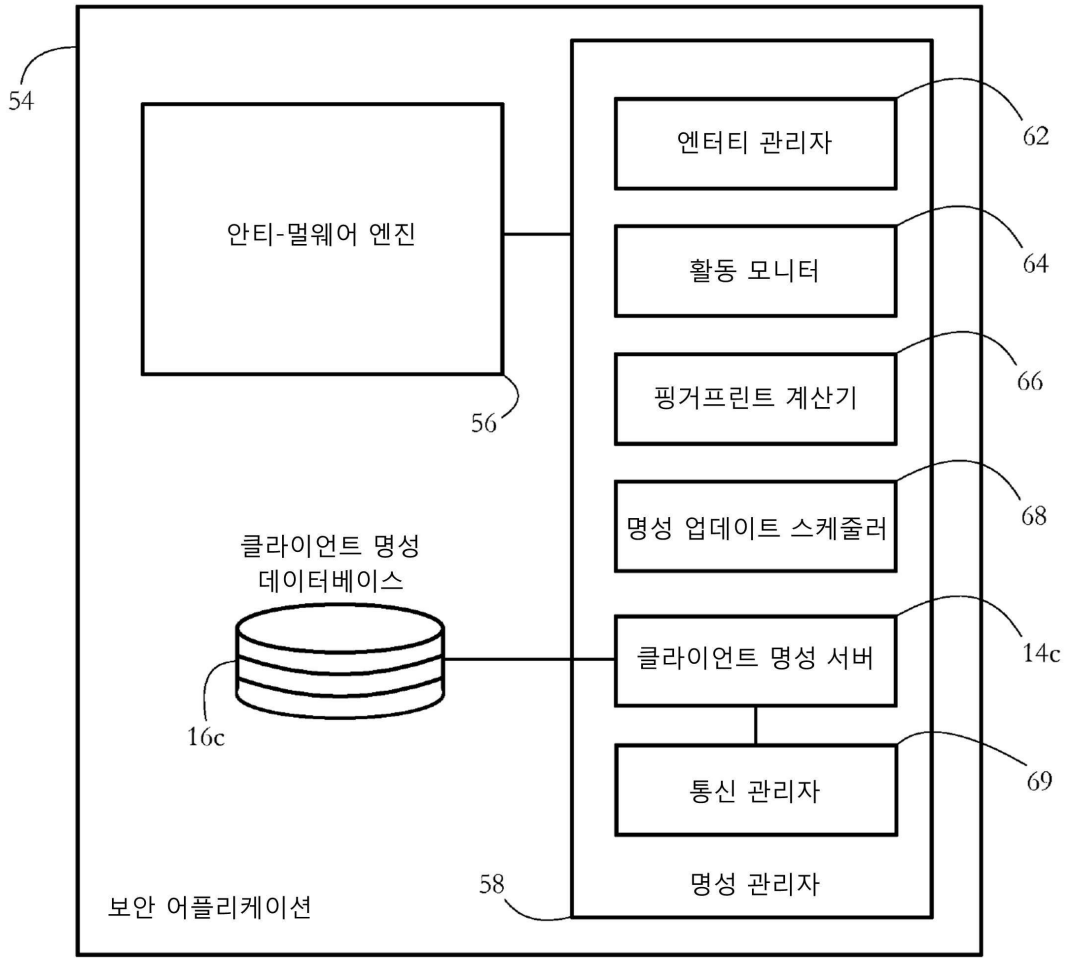
도면4b



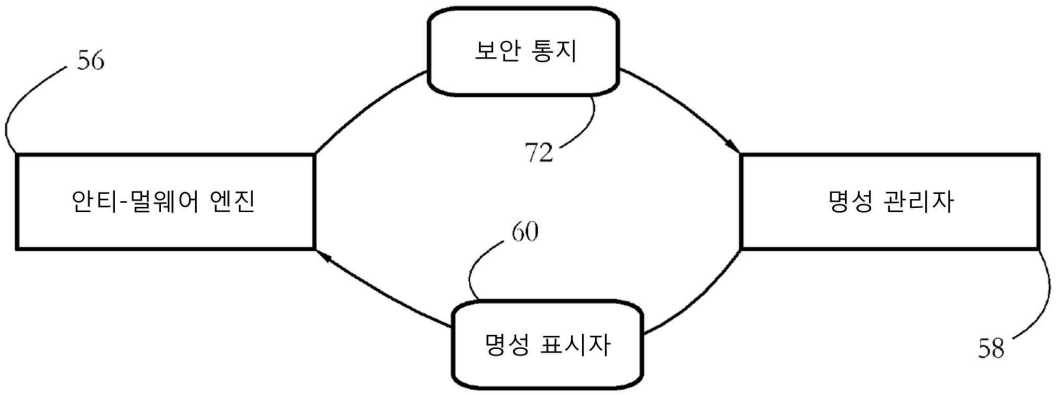
도면5


52a
52b
52c
54
어플리케이션
어플리케이션
어플리케이션
보안 어플리케이션
46
운영 시스템
클라이언트 시스템 하드웨어
10

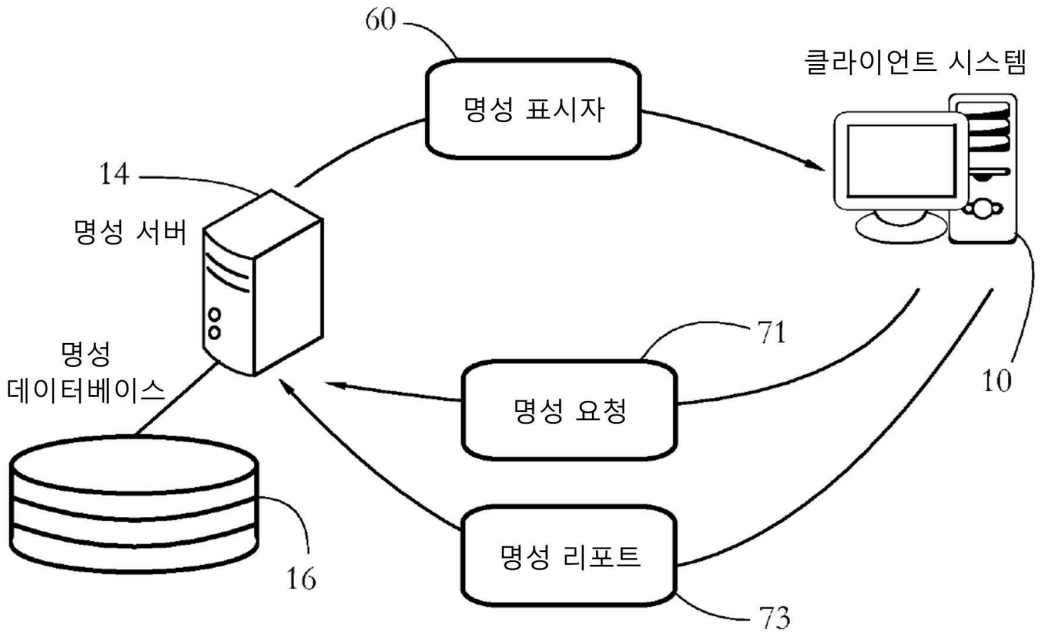
도면6


54
안티-멀웨어 엔진
56
엔터티 관리자
62
활동 모니터
64
핑거프린트 계산기
66
명성 업데이트 스케줄러
68
클라이언트 명성 데이터베이스
16c
클라이언트 명성 서버
14c
통신 관리자
69
58
명성 관리자
보안 어플리케이션

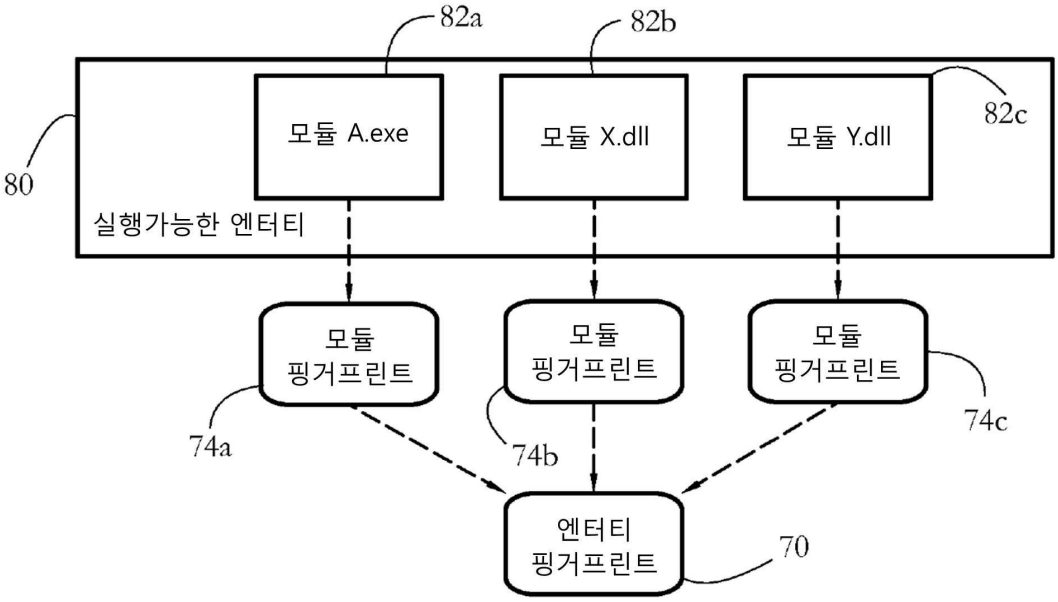
도면7


보안 통지
56
72
안티-멀웨어 엔진
명성 관리자
60
명성 표시자
58

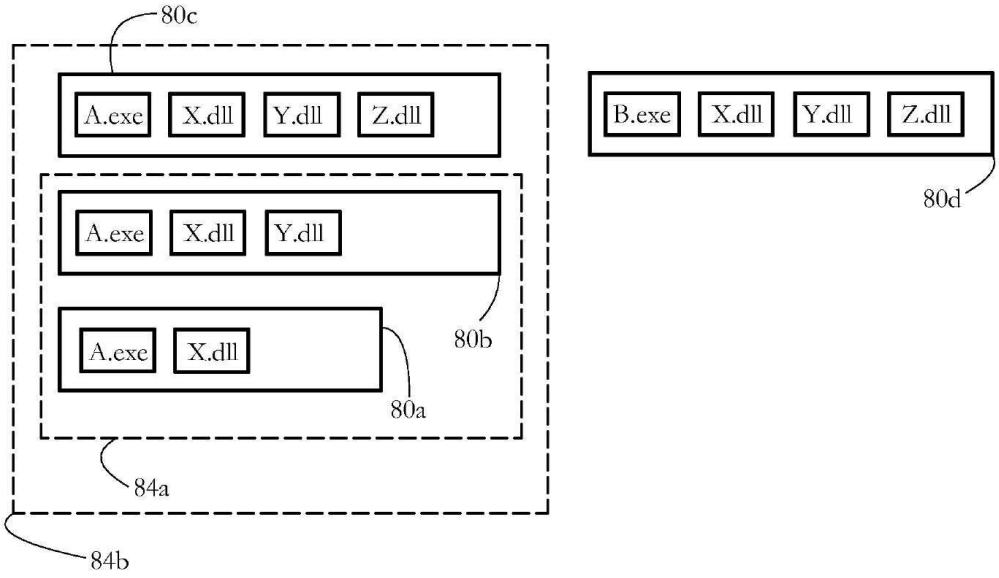
도면8


60
명성 표시자
클라이언트 시스템
14
명성 서버
71
명성
데이터베이스
명성 요청
10
명성 리포트
16
73

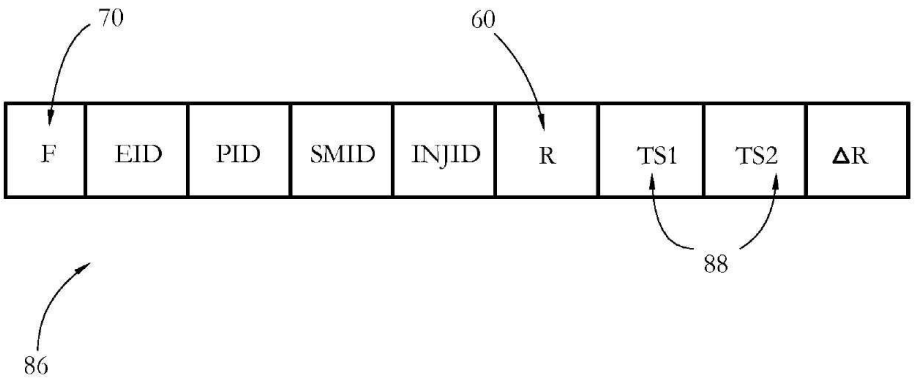
도면9



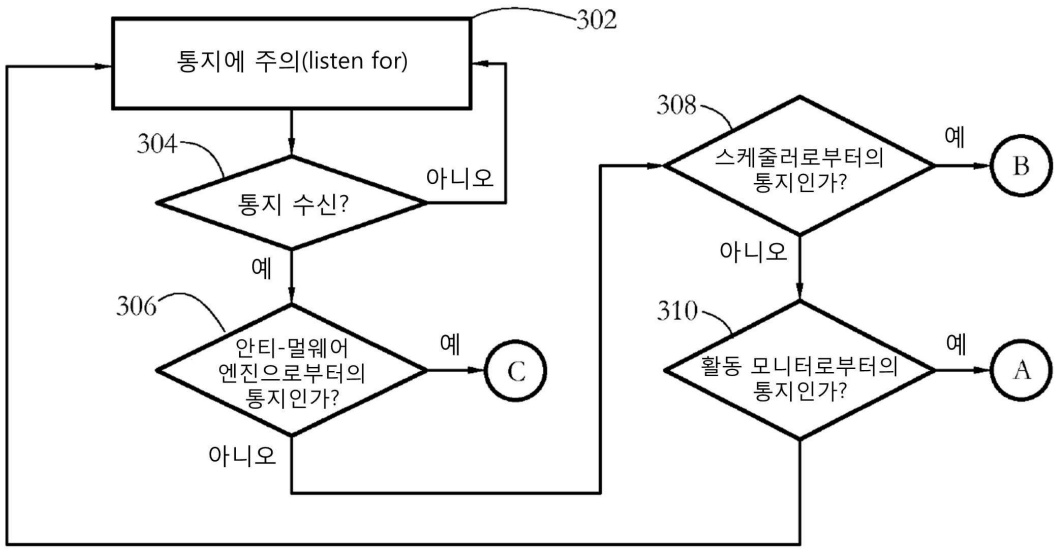
도면10



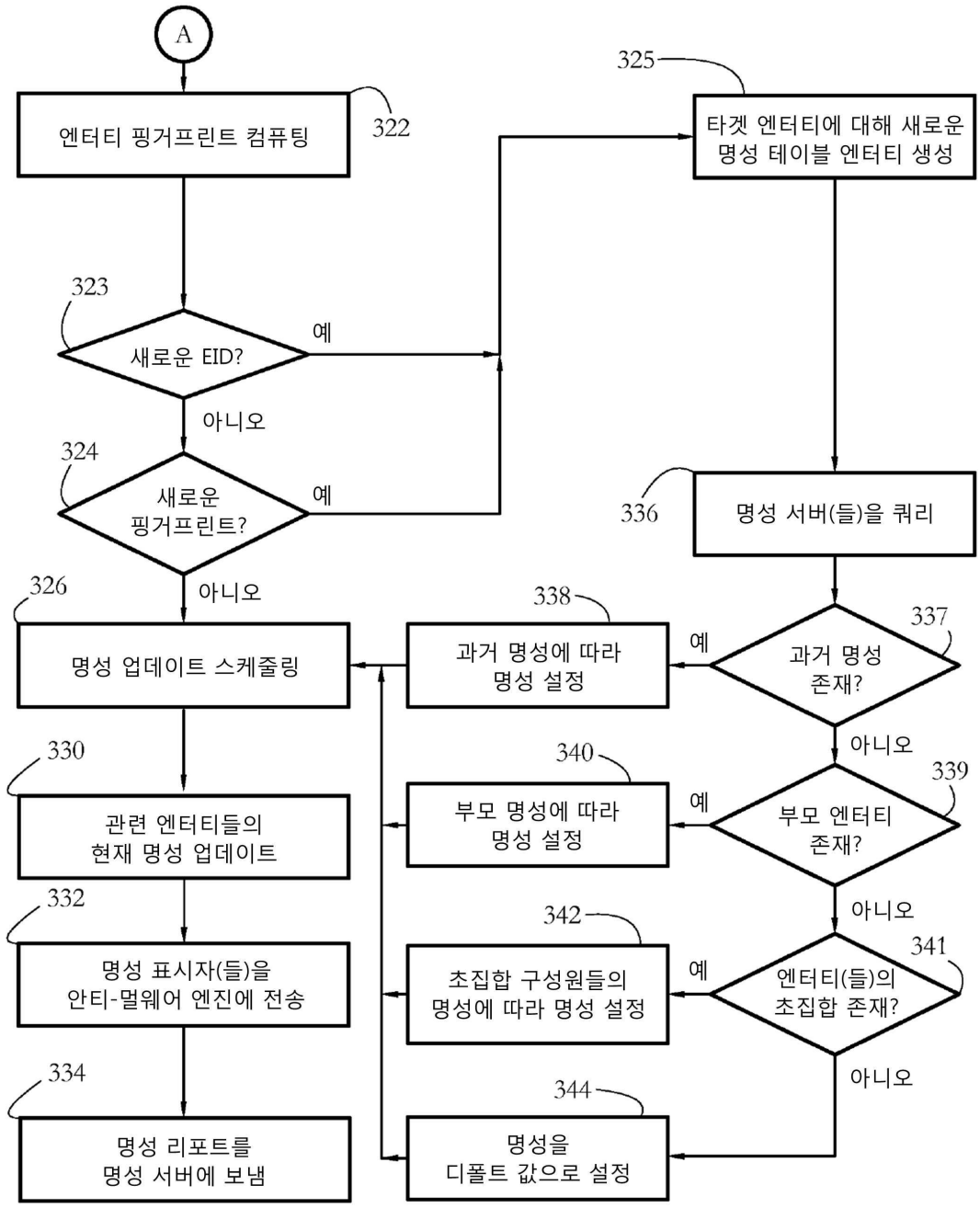
도면11



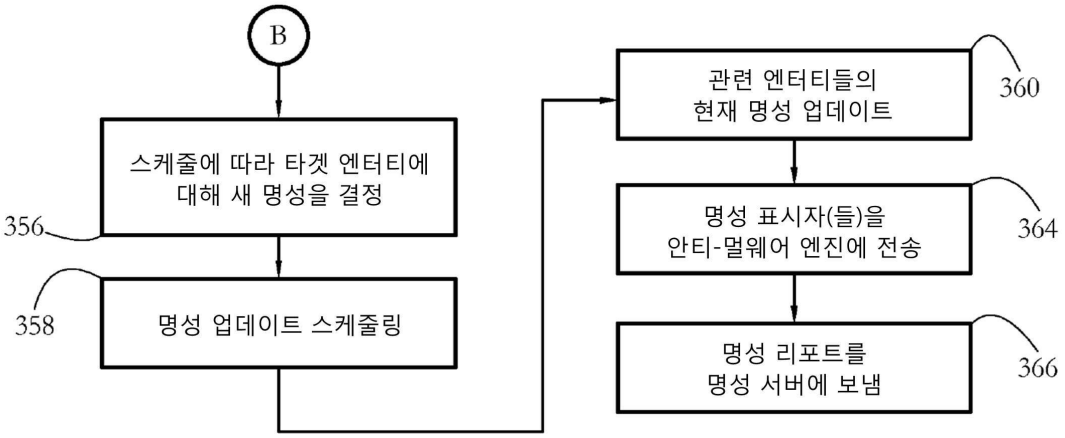
도면12a


302
통지에 주의(listen for)
304
통지 수신?
아니오
예
306
안티-멀웨어 엔진으로부터의 통지인가?
예
C
아니오
308
스케줄러로부터의 통지인가?
예
B
아니오
310
활동 모니터로부터의 통지인가?
예
A

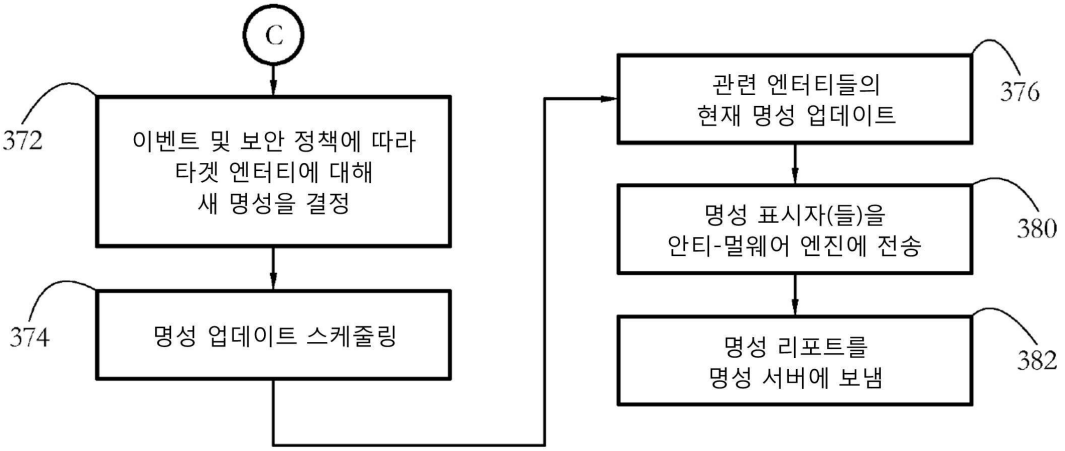
도면12b


A
엔터티 핑거프린트 컴퓨팅
322
323
새로운 EID?
예
아니오
324
새로운 핑거프린트?
예
아니오
325
타겟 엔터티에 대해 새로운 명성 테이블 엔터티 생성
336
명성 서버(들)을 쿼리
337
과거 명성 존재?
예
아니오
338
과거 명성에 따라 명성 설정
339
부모 엔터티 존재?
예
아니오
340
부모 명성에 따라 명성 설정
341
엔터티(들)의 초집합 존재?
예
아니오
342
초집합 구성원들의 명성에 따라 명성 설정
344
명성을 디폴트 값으로 설정
326
명성 업데이트 스케줄링
330
관련 엔터티들의 현재 명성 업데이트
332
명성 표시자(들)을 안티-멀웨어 엔진에 전송
334
명성 리포트를 명성 서버에 보냄

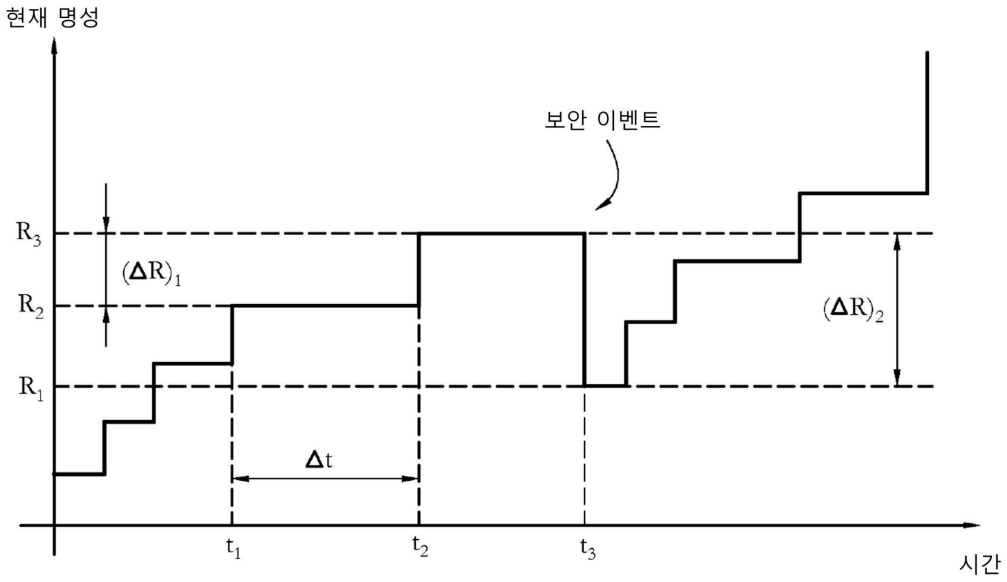
도면12c



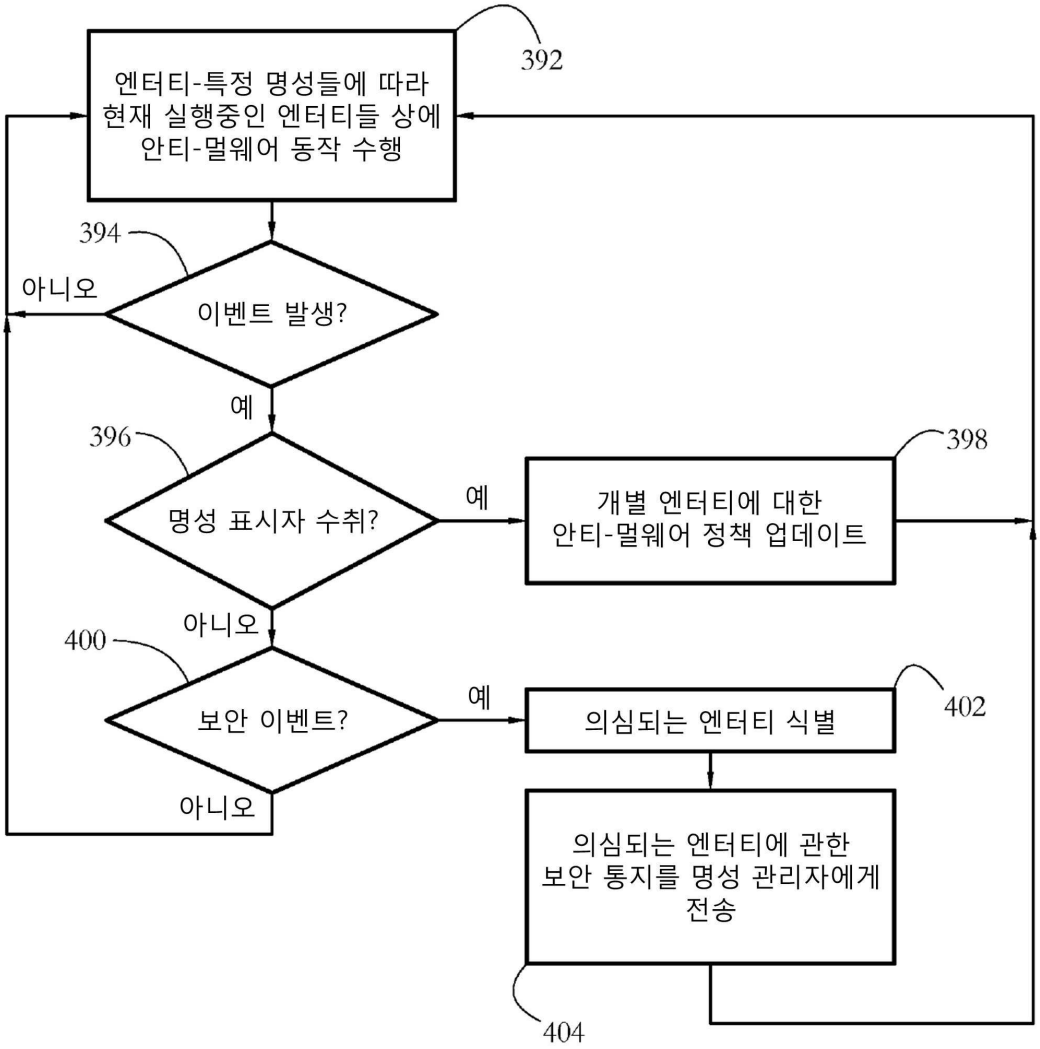
도면12d



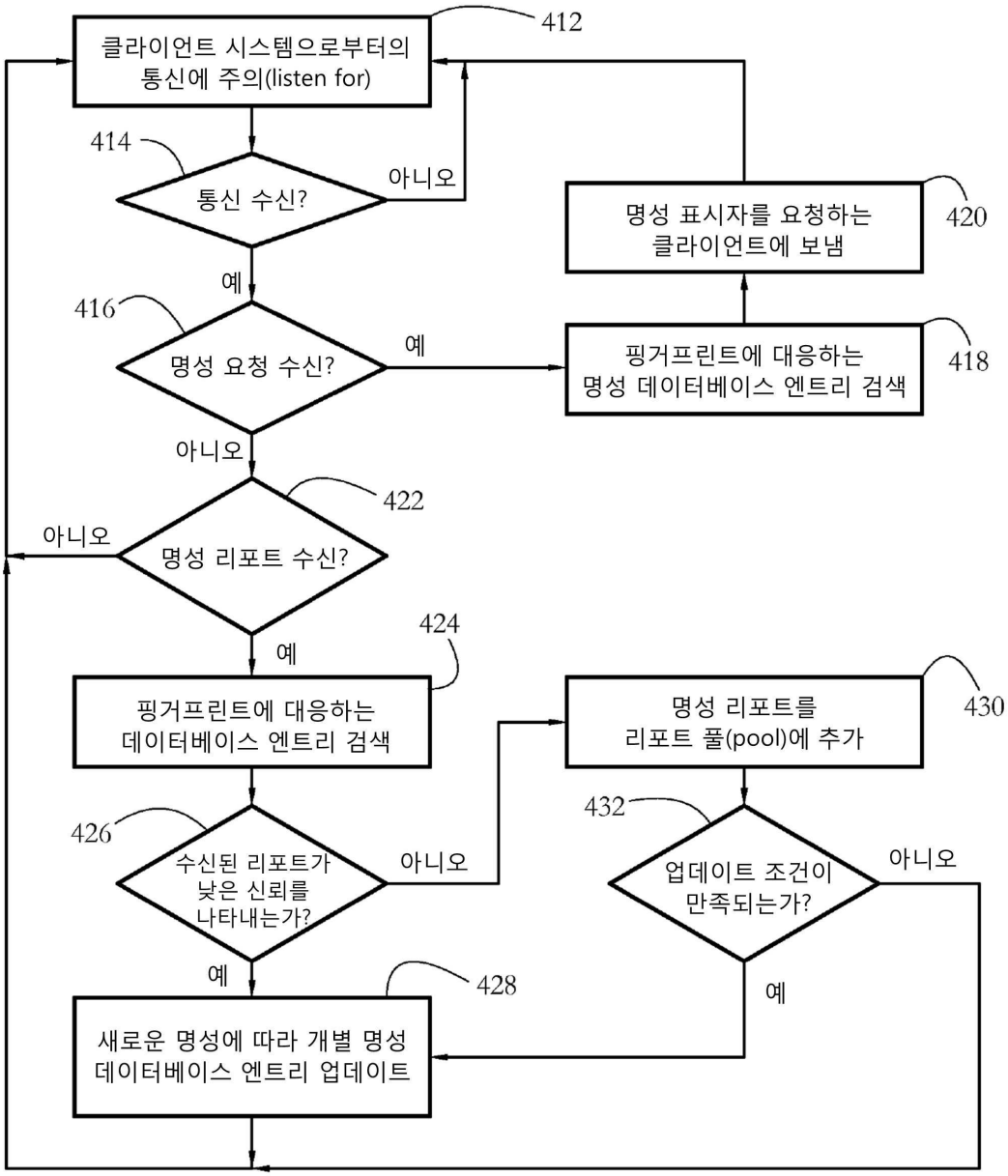
도면13



도면14


392
엔터티-특정 명성들에 따라 현재 실행중인 엔터티들 상에 안티-멀웨어 동작 수행
394
이벤트 발생?
아니오
예
396
명성 표시자 수취?
예
398
개별 엔터티에 대한 안티-멀웨어 정책 업데이트
아니오
400
보안 이벤트?
예
402
의심되는 엔터티 식별
아니오
의심되는 엔터티에 관한 보안 통지를 명성 관리자에게 전송
404

도면15



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 제12항

【변경전】

상기 제4 엔터티

【변경후】

상기 다른 엔터티

【직권보정 2】

【보정항목】 청구범위

【보정세부항목】 제1항

【변경전】

구성되며.

【변경후】

구성되며,