(19)대한민국특허청(KR)
(12) 공개특허공보(A)

(51) 。Int. Cl.
G06F 1/00 (2006.01)
G06F 15/00 (2006.01)
G06F 21/00 (2006.01)

(11) 공개번호 10-2006-0127007
(43) 공개일자 2006년12월11일

(21) 출원번호 10-2006-7012376
(22) 출원일자 2006년06월21일
심사청구일자 없음
번역문 제출일자 2006년06월21일
(86) 국제출원번호 PCT/IB2004/052674
국제출원일자 2004년12월06일
(87) 국제공개번호 WO 2005/064433
국제공개일자 2005년07월14일

(30) 우선권주장 03104884.6 2003년12월22일 유럽특허청(EPO)(EP)

(71) 출원인 코닌클리케 필립스 일렉트로닉스 엔.브이.
네델란드왕국, 아인드호펜, 그로네보드스베그 1

(72) 발명자 지달로프 니콜코
네덜란드, 아아 아인트호펜 엔엘-5656, 홀스트란 6 내

(74) 대리인 정상구
홍동오

전체 청구항 수 : 총 30 항

(54) 능동 엔티티를 사용하는 소프트웨어 실행 보호

(57) 요약

본 발명은 컴퓨터 프로그램 요소의 적어도 하나의 정적 리소스를 추출하는 단계(단계 102), 및 제 1 키(314)로 상기 정적 리소스를 암호화하는 단계(단계 106)를 포함하는, 상기 컴퓨터 프로그램 요소의 실행 보호를 가능하게 하기 위하여 적어도 일부의 컴퓨터 프로그램 요소를 암호화하는 단계, 및 제 1 엔티티에서 제 1 키로 암호화된 상기 정적 리소스(406)를 얻는 단계, 제 2 엔티티에 상기 암호화된 정적 리소스를 제공하는 단계(단계 208), 제 1 키로 암호화된 상기 정적 리소스(406)를 얻는 단계(단계 218), 제 2 키(422)를 얻는 단계(단계 216), 상기 제 2 키를 사용하여 상기 암호화된 정적 리소스를 복호화하는 단계(단계 222), 제 1 엔티티에 상기 정적 리소스를 제공하는 단계(단계 228), 및 상기 제 1 엔티티에 의해 상기 제 2 엔티티로부터 상기 정적 리소스를 얻는 단계(단계 210)를 포함하는, 상기 암호화된 정적 리소스를 복호화하는 단계에 관한 것이다.

대표도

도 4

특허청구의 범위

청구항 1.

컴퓨터 프로그램 요소의 실행 보호를 가능하게 하기 위하여 상기 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법에 있어서,

상기 컴퓨터 프로그램 요소의 적어도 하나의 정적 리소스들(306)을 추출하는 단계(단계 102); 및

키(314)로 적어도 하나의 정적 리소스(306)를 암호화하는 단계(단계 106)를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 2.

제 1 항에 있어서, 상기 컴퓨터 프로그램 요소에 상기 적어도 하나의 암호화된 정적 리소스(310)를 저장하는 단계(단계 110)를 더 포함하는, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 3.

제 1 항에 있어서, 상기 키(314)는 공개/개인 키 쌍 중 공개 키인, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 4.

제 3 항에 있어서, 컴퓨터 프로그램 요소에 상기 공개 키(314)를 저장하는 단계(단계 112)를 더 포함하는, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 5.

제 3 항에 있어서,

대응하는 개인 키(316)를 얻는 단계; 및

상기 컴퓨터 프로그램 요소가 제공된 엔티티로부터 독립된 엔티티(318)에서 상기 개인 키(316)를 저장하는 단계(단계 114)를 더 포함하는, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 6.

제 1 항에 있어서, 상기 추출 단계(단계 102)는 상기 프로그램 요소의 어떤 위치로부터 상기 적어도 하나의 정적 리소스(306)를 추출하는 단계를 포함하고 상기 저장 단계(단계 110)는 상기 위치에서 상기 암호화된 정적 리소스(310)를 저장하는 단계를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부분을 암호화하는 방법.

청구항 7.

컴퓨터 프로그램 요소의 실행 보호를 가능하게 하기 위하여 컴퓨터 프로그램 요소의 적어도 일부를 암호화하는 컴퓨터 프로그램 암호화 디바이스에 있어서,

상기 컴퓨터 프로그램 요소(302)의 적어도 하나의 정적 리소스(306)를 추출하고(단계 106); 및

키(314)로 적어도 하나의 정적 리소스들(306)을 암호화(단계 106)하도록 구성되는, 컴퓨터 프로그램 암호화 디바이스.

청구항 8.

컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 있어서,

상기 프로그램 코드 수단이 컴퓨터 내에 로딩될 때, 상기 프로그램 코드 수단은 상기 컴퓨터로 하여금,

상기 컴퓨터 프로그램 요소(302)의 적어도 하나의 정적 리소스(306)를 추출하는 단계(단계 102); 및

키(314)로 상기 적어도 하나의 정적 리소스(306)를 암호화(단계 106)하는 단계를 실행시키도록 하는, 컴퓨터 프로그램 제품.

청구항 9.

컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 있어서,

상기 컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 상기 프로그램 코드 수단은 상기 컴퓨터로 하여금,

상기 컴퓨터 프로그램 요소(302)의 적어도 하나의 정적 리소스(306)를 추출하는 단계(단계 102); 및

키(314)로 상기 적어도 하나의 정적 리소스(306)를 암호화하는 단계(단계 106)를 실행시키도록 하는, 컴퓨터 프로그램 요소.

청구항 10.

컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 있어서,

키(310)로 암호화된 적어도 하나의 정적 리소스를 포함하는, 컴퓨터 프로그램 제품.

청구항 11.

제 10 항에 있어서, 상기 키(314)는 공개/개인 키 쌍 중 공개 키인, 컴퓨터 프로그램 제품.

청구항 12.

제 11 항에 있어서, 상기 컴퓨터 프로그램 코드 수단은 상기 공개 키(314)를 더 포함하는, 컴퓨터 프로그램 제품.

청구항 13.

컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 있어서,

키(310)로 암호화된 적어도 하나의 정적 리소스를 포함하는, 컴퓨터 프로그램 요소.

청구항 14.

컴퓨터 프로그램 요소(402)의 실행을 가능하게 하기 위해 상기 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법에 있어서,

제 1 엔티티(52)에서 제 1 키(314)로 암호화된 적어도 하나의 정적 리소스(406)를 얻는 단계;

제 2 엔티티(54)에 상기 적어도 하나의 암호화된 정적 리소스(406)를 제공하는 단계(단계 208); 및

상기 제 1 엔티티(52)에 의해 제 2 엔티티(54)로부터 상기 적어도 하나의 정적 리소스(430)를 얻는 단계를 포함하고, 상기 제 1 키(314)에 따른 상기 암호화는 제 2 키(422)를 사용하여 복호화되는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 15.

제 14 항에 있어서,

제 3 키를 얻는 단계(단계 202); 및

상기 제 3 키를 사용하여 상기 적어도 하나의 암호화된 정적 리소스(430)를 복호화하는 단계(단계 212)를 더 포함하고,

제공 단계(단계 208)는 제 3 키(404) 및 상기 적어도 하나의 암호화된 정적 리소스(406, 410)를 상기 제 2 엔티티(54)에 제공하는 단계를 포함하고, 상기 제 1 엔티티(52)에 의해 상기 제 2 엔티티(54)로부터 상기 적어도 하나의 정적 리소스(430)를 얻는 단계(단계 210)는 컴퓨터 프로그램 요소가 실행되도록, 상기 제 3 키(426)로 암호화된 상기 적어도 하나의 정적 리소스(430)를 얻는 단계(430)를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 16.

제 14 항에 있어서, 상기 제 3 키(404, 432)는 랜덤 세션 키(random session key)인, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 17.

제 14 항에 있어서,

제 1 키(408)를 얻는 단계; 및

상기 제 1 키(408)를 사용하여 상기 제 3 키(404) 및 상기 적어도 하나의 암호화된 정적 리소스(406)를 암호화하는 단계를 포함하고(단계 206),

상기 적어도 하나의 암호화된 정적 리소스(410)를 상기 제 2 엔티티(54)에 제공하는 단계(단계 208)는 상기 제 1 키(408)를 사용하여 모두 암호화된(410) 상기 제 3 키(404) 및 상기 적어도 하나의 암호화된 정적 리소스(406)를 제공하는 단계를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 18.

제 14 항에 있어서, 상기 제 1 키(314, 408) 및 상기 제 2 키(316)는 공개/개인 키 쌍중 각각 공개 키 및 개인 키인, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 19.

컴퓨터 프로그램 요소의 실행을 가능하게 하기 위해 상기 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법에 있어서,

제 1 엔티티(52)로부터 적어도 하나의 암호화된 정적 리소스(414)를 얻는 단계(단계 218)로서, 적어도 하나의 정적 리소스(306)는 제 1 키(314)를 사용하여 암호화되는, 상기 얻는 단계(단계 218);

제 2 키(416)를 얻는 단계(단계 216);

상기 제 2 키(416)를 사용하여 적어도 하나의 암호화된 정적 리소스(418)를 복호화하는 단계(단계 222); 및

상기 제 1 엔티티(52)에 상기 적어도 하나의 정적 리소스(424)를 제공하는 단계(단계 228)를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 20.

제 19 항에 있어서,

상기 제 1 엔티티(52)로부터 제 3 키(420)를 얻는 단계; 및

상기 제 3 키(426)를 사용하여 상기 적어도 하나의 정적 리소스(424)를 암호화하는 단계를 더 포함하고,

상기 적어도 하나의 정적 리소스(428)를 상기 제 1 엔티티(52)에 제공하는 단계(단계 228)는 상기 제 3 키(426)로 암호화된 상기 적어도 하나의 정적 리소스(428)를 제공하는 단계를 포함하는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 21.

제 20 항에 있어서, 상기 적어도 하나의 암호화된 정적 리소스(406) 및 상기 제 3 키(404)는 암호화(414)되어 얻어지고, 상기 암호화는 제 1 키(314)를 사용하여 이루어지는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 22.

제 21 항에 있어서, 상기 제 2 키(416)를 사용하여 상기 암호화된(414) 적어도 하나의 암호화된 정적 리소스(406) 및 제 3 키(404)를 복호화하는 단계(단계 220)를 더 포함하는, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 23.

제 19 항에 있어서, 상기 제 1 키(314) 및 제 2 키(416, 422)는 공개/개인 키 쌍중 각각 공개 키 및 개인 키인, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 24.

제 19 항에 있어서, 상기 제 3 키(420, 426)는 랜덤 세션 키인, 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법.

청구항 25.

컴퓨터 프로그램 요소를 실행하기 위하여 상기 컴퓨터 프로그램 요소(402)의 적어도 일부를 복호화하는 컴퓨터 프로그램 복호화 디바이스(52)에 있어서,

제 1 키(314)로 암호화된 적어도 하나의 정적 리소스(406)를 얻고,

상기 적어도 하나의 암호화된 정적 리소스(406)를 제 2 엔티티(54)에 제공하고(단계 208), 및

상기 제 2 엔티티(54)로부터 상기 적어도 하나의 정적 리소스(430)를 얻도록(단계 210) 구성되고, 제 1 키(314)에 따른 상기 암호화는 제 2 키(422)를 사용하여 복호화되는, 컴퓨터 프로그램 복호화 디바이스.

청구항 26.

컴퓨터 프로그램 요소를 실행하기 위하여 상기 컴퓨터 프로그램 요소의 적어도 일부를 복호화하기 위한 컴퓨터 프로그램 복호화 디바이스(54)에 있어서,

제 1 엔티티(52)로부터 적어도 하나의 암호화된 정적 리소스(414)를 얻고(단계 218), 적어도 하나의 정적 리소스(414)는 제 1 키(314)를 사용하여 암호화되고;

제 2 키(416)를 얻고(단계 216);

상기 제 2 키(422)를 사용하여 상기 적어도 하나의 암호화된 정적 리소스(418)를 복호화하고(단계 222);

상기 제 1 엔티티(52)에 상기 적어도 하나의 정적 리소스(424)를 제공(단계 228)하도록 구성된, 컴퓨터 프로그램 복호화 디바이스.

청구항 27.

컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터를 실행하게 하기 위하여 컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 있어서,

제 1 엔티티(52)에서 제 1 키(314)로 암호화된 적어도 하나의 정적 리소스(406)를 얻는 단계;

제 2 엔티티(54)에 상기 적어도 하나의 암호화된 정적 리소스(406)를 제공하는 단계(단계 208); 및

상기 제 1 엔티티(52)에 의해 상기 제 2 엔티티(54)로부터 상기 적어도 하나의 정적 리소스(430)를 얻는 단계(단계 210)를 포함하고, 상기 제 1 키(314)에 따른 상기 암호화는 제 2 키(422)를 사용하여 복호화되는, 컴퓨터 프로그램 제품.

청구항 28.

컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 상기 컴퓨터를 실행하게 하는 상기 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 있어서,

제 1 엔티티(52)에서 제 1 키(314)로 암호화된 적어도 하나의 정적 리소스(406)를 얻는 단계;

제 2 엔티티(54)에 상기 적어도 하나의 암호화된 정적 리소스(406)를 제공하는 단계(단계 208); 및

상기 제 1 엔티티(52)에 의해 상기 제 2 엔티티(54)로부터 상기 적어도 하나의 정적 리소스(430)를 얻는 단계를 포함하고, 상기 제 1 키(314)에 따른 상기 암호화는 제 2 키(422)를 사용하여 복호화되는, 컴퓨터 프로그램 요소.

청구항 29.

컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 있어서,

상기 프로그램 코드 수단이 컴퓨터에 로딩될 때, 상기 프로그램 코드 수단은 상기 컴퓨터로 하여금,

제 1 엔티티(52)로부터 적어도 하나의 암호화된 정적 리소스(414)를 얻는 단계(단계 218)로서, 상기 적어도 하나의 정적 리소스(414)는 제 1 키(314)를 사용하여 암호화되는, 상기 얻는 단계(218);

제 2 엔티티(54)에서 제 2 키(416)를 얻는 단계(단계 216);

상기 제 2 키(422)를 사용하여 상기 적어도 하나의 암호화된 정적 리소스(418)를 복호화하는 단계(단계 222); 및

상기 제 1 엔티티(52)에 상기 적어도 하나의 정적 리소스(424)를 제공하는 단계(단계 228)를 실행시키도록 하는, 컴퓨터 프로그램 제품.

청구항 30.

컴퓨터를 실행하게 하기 위하여 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 있어서,

제 1 엔티티(52)로부터 적어도 하나의 암호화된 정적 리소스(414)를 얻는 단계(단계 218)로서, 상기 적어도 하나의 정적 리소스(414)는 제 1 키(314)를 사용하여 암호화되는, 상기 얻는 단계(218);

제 2 엔티티(54)에서 제 2 키(416)를 얻는 단계(단계 216);

상기 제 2 키(422)를 사용하여 상기 적어도 하나의 암호화된 정적 리소스(418)를 복호화하는 단계(단계 222); 및

상기 제 1 엔티티(52)에 상기 적어도 하나의 정적 리소스(424)를 제공하는 단계(단계 228)를 포함하는, 컴퓨터 프로그램 요소.

명세서

기술분야

본 발명은 일반적으로 컴퓨터 프로그램 코드의 실행을 방지하는 것, 특히 능동 엔티티를 사용함으로써 정적 데이터를 암호화 및 복호화하는 것에 관한 것이다.

배경기술

강력한 실행 보호 방법들은 예를 들어 PC(퍼스널 컴퓨터)의 예를 들어, USB(유니버셜 시리얼 버스) 포트 또는 프린터 포트 같은 병렬 또는 직렬 포트에 접속되는 하나의 실시예로서 한가지 형태의 능동 엔티티인 소위 하드웨어 동글(dongle)을 이용한다. 동글은 통상적으로 수동 요소이지만 몇몇 암호화/복호화 키들이 로딩된 프로그램 가능 메모리를 포함할 수 있다. 정보는 PC 및 동글 사이에서 교환될 수 있다. 상기 동글은 예를 들어 다음 두 개의 방식으로 사용될 수 있다.

1. 소프트웨어 실행의 보호를 위하여, 쉘(shell) 프로그램은 보호될 소프트웨어 주변에 생성된다. 쉘을 생성하는 과정에서, 본래 소프트웨어는 동글의 키들에 따라 완전히 또는 부분적으로 암호화되고, 그후 암호화는 쉘에 삽입된다. 따라서 생성된 쉘은 동글로부터의 키들을 바탕으로 하지만 소프트웨어를 복호화하기 위하여 사용되는 알고리즘을 바탕으로 한다. 쉘이 시작될 때, 동글로부터의 키들을 검색하고, 암호화된 소프트웨어를 추출하고, 상기 암호화된 소프트웨어를 복호화하고 본래 소프트웨어를 운용한다. 동글이 제공되지 않거나 다른 키들을 포함하는 다른 동글이 사용되는 경우, 복호화는 실패한다.

2. 또한, 소프트웨어의 실행 보호를 위하여, 본래 프로그램의 진입점은 하나의 과정의 진입점으로 대체될 수 있다. 논리 기능은 제공되고 동글로부터 키들을 검색한다. 검색 키들을 바탕으로, 복합 논리는 동글이 올바른 동글인지, 또는 아닌지의 여부를 결정하기 위하여 구성된다. 성공적인 동글 식별 후, 기능은 본래 소프트웨어의 실행을 가능하게 하는 본래 프로그램 진입점을 호출한다.

그러나 상기된 방법들에는 몇가지 단점이 있다.

PC 및 동글 사이의 다른 통신 세션들의 통신 콘텐트는 일반적으로 동일하고, 이것은 상기 통신을 도청함으로써, 프로토콜 및 키들을 검색하고, 추후 본래 동글을 요구하지 않고 하드웨어 또는 소프트웨어의 동글을 대리 실행할 수 있는 것을 의미한다.

동글이 식별된 후, 본래 프로그램의 진입점은 호출되고, 본래 프로그램은 본래 메모리에 제공된다. 경험이 있는 사용자는 상기 프로그램을 실행할 수 있는 휴대기에 상기 프로그램을 다시 기입할 수 있다.

리버스 논리를 사용하여 쉽게 대체될 수 있는 동글 검사 코드의 if 명령들이 일반적으로, 하나 또는 몇 개가 있다.

일반적으로 보안 툴들 및 하드웨어 동글들은 지. 하체즈(G. Hachez), "전자 상거래에 적합한 소프트웨어 보호 툴들과 소프트웨어 워터마킹 및 스마트 카드들에의 기여와의 비교 연구(A comparative study of software protection tools suited for E-commerce with contributions to software watermarking and smart cards)" 박사 논문, UCL, 루베인-라-뉴브(Louvain-La-Neuve), 벨기에, 3월, 2003에 기술된다.

이 서류에 따라, 최신 하드웨어 동글 버젼들은 USB 포트에 플러그 접속되고 일반적으로 스마트 카드의 CPU에 삽입된다. 이들 버젼들은 각각의 챌린지(challenge)에 다른 값을 리턴할 작은 마이크로 제어기를 포함한다. 소프트웨어는 챌린지를 가진 동글에게 정기적으로 질문할 것이고 대답이 옳다는 것을 검증할 것이다. 대부분의 진보된 동글들은 작은 양의 메모리를 가진 작은 마이크로 제어기를 포함한다. 이 경우, 소프트웨어의 몇몇 중요 부분들은 동글 내에서 실행된다.

상기된 하드웨어 동글 버젼들의 방법은 다음 단점들을 가진다. 첫째, 보호될 소프트웨어 내의 동글의 모든 검사들을 제거하는 것이 성공될 수 있다는 위험성이 있다. 둘째, 동글이 침입자에 의해 대리 실행되는 위험성이 있다.

따라서 소프트웨어의 동글 같은 능동 엔티티에 대한 검사의 제거 후 조차 소프트웨어가 운용될 수 없는 소프트웨어 실행 보호 방법이 필요하다. 추가로 올바른 엔티티가 제공되는지 또는 아닌지의 여부에 따라 단일 if-then 명령들을 포함하지 않는 방법이 필요하다.

발명의 상세한 설명

본 발명의 목적은 상기 컴퓨터 프로그램 요소의 정적 리소스들의 암호화를 사용함으로써 컴퓨터 프로그램 요소의 실행 보호를 제공하는 것이다.

본 발명의 제 1 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행보호를 가능하게 하기 위하여 컴퓨터 프로그램 요소의 적어도 일부를 암호화하는 방법에 의해 달성되고, 상기 방법은,

상기 컴퓨터 프로그램 요소의 적어도 하나의 정적 리소스를 추출하는 단계, 및

하나의 키로 적어도 하나의 정적 리소스를 암호화하는 단계를 포함한다.

본 발명의 제 2 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행 보호를 가능하게 하기 위해 컴퓨터 프로그램 요소의 적어도 일부를 암호화하기 위한 컴퓨터 프로그램 암호화 디바이스에 의해 달성되고, 상기 디바이스는,

상기 컴퓨터 프로그램 요소의 적어도 하나의 정적 리소스를 추출하고, 및

하나의 키로 적어도 하나의 정적 리소스를 암호화하도록 구성된다.

본 발명의 제 3 측면에 따라, 이 목적은, 컴퓨터 프로그램 코드 수단으로서, 컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터로 하여금,

상기 컴퓨터 프로그램 요소의 적어도 하나의 정적 리소스를 추출하는 단계, 및

하나의 키로 적어도 하나의 정적 리소스를 암호화하는 단계를 실행시키도록하는, 상기 컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 의해 달성된다.

본 발명의 제 4 측면에 따라, 이 목적은, 컴퓨터 프로그램 코드 수단으로서, 컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터로 하여금,

상기 컴퓨터 프로그램 요소의 적어도 하나의 정적 소스를 추출하는 단계, 및

하나의 키로 적어도 하나의 정적 리소스를 암호화하는 단계를 실행시키도록 하는, 상기 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 의해 달성된다.

본 발명의 제 5 측면에 따라, 이 목적은 그 위에 컴퓨터 프로그램 코드 수단을 가진 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 의해 달성되고, 상기 제품은,

하나의 키로 암호화된 적어도 하나의 정적 리소스를 포함한다.

본 발명의 제 6 측면에 따라, 이 목적은 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 의해 달성되고, 상기 요소는,

하나의 키로 암호화된 적어도 하나의 정적 리소스를 포함한다.

본 발명의 제 7 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행을 가능하게 하기 위한 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법에 달성되고, 상기 방법은,

제 1 엔티티에서 제 1 키로 암호화된 적어도 하나의 정적 리소스를 얻는 단계,

제 2 엔티티에 적어도 하나의 상기 암호화된 정적 리소스를 제공하는 단계, 및

제 1 엔티티에 의해 제 2 엔티티로부터 상기 적어도 하나의 정적 리소스를 얻는 단계를 포함하고, 제 1 키에 따른 암호화는 제 2 키를 사용하여 복호화된다.

본 발명의 제 8 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행을 가능하게 하기 위하여 컴퓨터 프로그램 요소의 적어도 일부를 복호화하는 방법에 의해 달성되고, 상기 방법은,

적어도 하나의 정적 리소스가 제 1 키를 사용하여 암호화되는, 적어도 하나의 암호화된 정적 소스를 제 1 엔티티로부터 얻는 단계,

제 2 키를 얻는 단계,

상기 제 2 키를 사용함으로써 상기 적어도 하나의 암호화된 정적 리소스를 복호화하는 단계, 및

제 1 엔티티에 상기 적어도 하나의 정적 리소스를 제공하는 단계를 포함한다.

본 발명의 제 9 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행을 가능하게 하기 위한 컴퓨터 프로그램 요소의 적어도 일부를 복호화하기 위해 컴퓨터 프로그램 복호화 디바이스에 의해 달성되고, 상기 디바이스는,

제 1 키로 암호화된 적어도 하나의 정적 리소스를 얻고,

제 2 엔티티에 상기 적어도 하나의 암호화된 정적 리소스를 제공하고, 및

상기 적어도 하나의 정적 리소스를 제 2 엔티티로부터 얻도록 구성되고, 제 1 키에 따른 암호화는 제 2 키를 사용하여 복호화된다.

본 발명의 제 10 측면에 따라, 이 목적은 상기 컴퓨터 프로그램 요소의 실행을 가능하게 하기 위하여 컴퓨터 프로그램 요소의 적어도 일부를 복호화하기 위해 컴퓨터 프로그램 복호화 디바이스에 의해 달성되고, 상기 디바이스는,

적어도 하나의 정적 리소스가 제 1 키를 사용하여 암호화되는 적어도 하나의 암호화된 정적 리소스를 제 1 엔티티로부터 얻고,

제 2 키를 얻고,

상기 제 2 키를 사용하여 적어도 하나의 암호화된 정적 리소스를 복호화하고, 및

제 1 엔티티에 상기 적어도 하나의 정적 리소스를 제공하도록 구성된다.

본 발명의 제 11 측면에 따라, 이 목적은 또한, 컴퓨터 프로그램 코드 수단으로서, 상기 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터로 하여금,

제 1 엔티티에서 제 1 키로 암호화된 적어도 하나의 정적 리소스를 얻는 단계,

제 2 엔티티에 상기 적어도 하나의 암호화된 정적 리소스를 제공하는 단계, 및

상기 제 1 엔티티에 의해 제 2 엔티티로부터 상기 적어도 하나의 정적 리소스를 얻는 단계를 실행시키도록 하는, 상기 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 의해 달성되고, 여기서, 제 1 키에 따른 암호화는 제 2 키를 사용하여 복호화된다.

본 발명의 제 12 측면에 따라, 이 목적은 컴퓨터 프로그램 코드 수단으로서, 상기 컴퓨터 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터로 하여금,

제 1 엔티티에서 제 1 키로 암호화된 적어도 하나의 정적 리소스를 얻는 단계,

제 2 엔티티에 상기 적어도 하나의 암호화된 정적 리소스를 제공하는 단계, 및

제 1 엔티티에 의해 제 2 엔티티로부터 상기 적어도 하나의 정적 리소스를 얻는 단계를 실행시키도록 하는, 상기 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 의해 달성된다.

본 발명의 제 13 측면에 따라, 이 목적은 컴퓨터 프로그램 코드 수단으로서, 상기 프로그램 코드 수단이 컴퓨터에 로딩될 때, 컴퓨터로 하여금,

적어도 하나의 정적 리소스가 제 1 키를 사용하여 암호화되는, 적어도 하나의 암호화된 정적 리소스를 제 1 엔티티로부터 얻는 단계,

제 2 엔티티에서 제 2 키를 얻는 단계,

상기 제 2 키를 사용하여 상기 적어도 하나의 암호화된 정적 리소스를 복호화하는 단계, 및

제 1 엔티티에 상기 적어도 하나의 정적 리소스를 제공하는 단계를 실행시키도록 하는, 상기 컴퓨터 프로그램 코드 수단을 갖는 컴퓨터 판독 가능 매체를 포함하는 컴퓨터 프로그램 제품에 의해 달성된다.

본 발명의 제 14 측면에 따라, 이 목적은 컴퓨터를 실행하기 위한 컴퓨터 프로그램 코드 수단을 포함하는 컴퓨터 프로그램 요소에 의해 달성되고, 상기 요소는,

적어도 하나의 정적 리소스가 제 1 키를 사용하여 암호화되는, 적어도 하나의 암호화된 정적 리소스를 제 1 엔티티로부터 얻고,

제 2 엔티티에서 제 2 키를 얻고,

상기 제 2 키를 사용함으로써 상기 적어도 하나의 암호화된 정적 리소스를 복호화하고, 및

제 1 엔티티에 상기 적어도 하나의 정적 리소스를 제공한다.

본 발명의 배후에서 일반적인 생각은 상기 컴퓨터 프로그램 코드 내 정적 리소스에 대해 컴퓨터 프로그램 요소의 암호화를 사용함으로써 컴퓨터 프로그램 코드의 실행을 보호하는 것이다. 상기 생각은 추가로 상기 암호화된 정적 리소스의 복호화 동안 두 개의 엔티티들의 사용에 의존하고, 상기 두 개의 엔티티들 사이의 통신은 적어도 부분적으로 암호화된다.

본 발명은 다음 장점을 가진다.

1. 상기 컴퓨터 프로그램 코드의 실행에 중요한 적어도 하나의 정적 리소스를 암호화함으로써 컴퓨터 프로그램 코드의 실행 보호를 제공한다.

2. 복호화 처리는 제 1 및 제 2 엔티티를 요구한다.

3. 컴퓨터 프로그램 코드는 제 2 엔티티에 대한 요구들을 제거한 후조차도 제 1 엔티티 내에서 실행될 수 없다.

종속항들 및 상기 종속항들의 장점의 방향은 다음과 같다.

청구항 제 2 항은 상기 컴퓨터 요소에 적어도 하나의 암호화된 정적 리소스를 저장하는 것에 관한 것이다. 이 청구항은 컴퓨터 프로그램 요소의 실행 동안 필요한 리소스들이 암호화될 수 있는 장점을 가진다.

청구항 제 3 항, 제 11 항, 제 18 항 및 제 23 항은 공개/개인 키 쌍의 공개 키 및 개인 키를 사용하는 것에 관한 것이고, 그 장점은 하나의 키가 다른 키에 의해 암호화된 데이터를 복호화하기 위하여 필요하다는 것이다.

청구항 제 4 항 및 제 12 항은 각각 컴퓨터 프로그램 요소 및 컴퓨터 프로그램 코드 수단의 공개 키를 가지는 것에 관한 것이다. 이들 청구항들은 공개 키를 사용하여 암호화된 데이터를 복호화하기 위하여 보안 개인 키를 사용할 수 있는 장점을 가진다.

청구항 제 5 항은 공개 키에 대응하는 개인 키를 얻고, 컴퓨터 프로그램 요소가 제공된 엔티티와 별개의 엔티티에 상기 개인 키를 저장하는 것에 관한 것이다. 이 청구항은 두 개의 엔티티들을 분리를 가능하게 함으로써 실행 보호 보안을 크게 향상시키는 장점을 가진다.

청구항 제 6 항은 컴퓨터 프로그램 요소의 위치로부터 적어도 하나의 정적 리소스를 추출하고 상기 위치에 암호화된 리소스를 저장하는 것에 관한 것이다. 이것은 첫째 본래 정보가 이용되지 않고 둘째, 다른 부분 또는 요소가 암호화된 리소스를 저장함으로써 영향받지 않는 장점을 가진다.

청구항 제 15 항 및 제 20 항은 제 3 키를 얻고 상기 제 3 키를 사용함으로써 적어도 하나의 정적 리소스를 암호화/복호화하는 것에 관한 것이다. 이들 청구항들은 하나의 엔티티에 의해 다른 엔티티에 보내진 정적 리소스가 상기 제 3 키로 암호화될 수 있는 장점을 가진다.

청구항 제 16 항 및 제 24 항은 랜덤 세션 키인 제 3 키를 사용하는 것에 관한 것이다. 대칭적인 키를 가지는 것의 장점은 동일한 키가 암호화 및 복호화에 사용될 수 있고, 이것은 사용되는 키들의 수를 제한시킨다.

청구항 제 17 항, 제 21 항 및 제 22 항은 제 3 키를 암호화/복호화하기 위한 제 1 키 및 적어도 하나의 암호화된 정적 리소스를 사용하는 것에 관한 것이다. 이것은 제 3 키가 하나의 엔티티로부터 다른 엔티티로 암호화되어 보내질 수 있는 장점을 가져서, 상기 제 3 키를 사용하여 정적 데이터의 암호화 보안을 향상시킬 수 있다.

본 발명의 이들 및 다른 측면들은 이후에 기술되는 실시예를 참조하여 명백하게 설명될 것이다.

이 명세서에 사용된 용어 "포함하다"는 상기된 특징들, 완전체들, 단계들 또는 구성요소들의 존재를 지정하기 위하여 취해지지만, 하나 이상의 다른 특징들, 완전체들, 단계들, 구성요소들 또는 그 그룹들의 존재 또는 부가를 배제하지 않는 것이 강조된다.

본 발명은 첨부된 도면들과 관련하여 판독되는 본 발명의 바람직한 실시예들의 다음 설명으로부터 보다 명확하게 이해될 것이다.

실시예

본 발명은 상기 컴퓨터 프로그램 코드의 정적 리소스들을 암호화 및 복호화함으로써 컴퓨터 프로그램 코드의 실행 보호에 관한 것이다.

암호화 및 복호화는 공용 키 암호계 아키텍쳐를 사용하고 보호될 컴퓨터 프로그램 코드의 소스 코드에 액세스하는 것을 요구한다.

본 발명의 하나의 실시예에 따라, 두 개의 다른 엔티티들은 암호화된 정보를 복호화하기 위한 처리에 사용된다. 도 5는 이들 두 개의 다른 엔티티들의 본 발명의 하나의 실시예를 도시한다. 퍼스널 컴퓨터(52) 같은 컴퓨터는 제 1 엔티티 및 능동 동글(54)을 도시하고, 제 2 엔티티를 도시한다. 이들 두 개의 엔티티들은 상기 처리의 복호화 단계들 동안 정보를 전송/수신하기 위하여 구성된다.

동글 대신, 보안 칩은 사용될 수 있다. 이런 보안 칩은 컴퓨터 플랫폼에 집적될 수 있다.

능동 동글에는 통상적으로 간단한 대칭 및 비대칭 암호화/복호화 알고리즘들을 운용할 수 있는 작은 처리기가 설치된다. 두 개의 엔티티들, 여기서 컴퓨터 및 능동 동글 사이의 인터페이스는 USB(유니버셜 시리얼 포트), 네트워크, 또는 다른 통신 채널일 수 있다. 컴퓨터 및 능동 동글 사이의 통신은 클라이언트-서버 모델을 바탕으로 한다.

보호될 컴퓨터 프로그램 코드가 시장에서 배분되기 전에, 정적 데이터의 적어도 일부는 추출되고 능동 동글의 공개 키를 사용하여 암호화되고, 암호화된 데이터로서 소스 코드로 대체된다. 컴퓨터 프로그램 코드를 컴파일 및 운용하는 중에, 대응하는 개인 키를 가진 동글만이 컴퓨터 프로그램 코드의 데이터를 사용하기 전에 데이터를 복호화할 수 있다.

하기된 바와 같이, 암호화는 통상적으로 상기 컴퓨터와 무관한 다른 곳에서 수행된다. 당업자에게 잘 알려진 바와 같이 정보의 암호화 및 복호화는 키 및 록(lock)과 유사한 방식으로 서로 관련된다. 여기서, 암호화의 과정은 컴퓨터 및 동글 사이의 통신 채널이 암호화된 데이터를 복호화하기 위하여 설정되도록 수행된다.

본 발명의 하나의 실시예에 따라, 컴퓨터 내의 복호화 시작 과정은 복호화된 프로그램 코드가 로딩되고, 통신 채널을 통하여 동글로 정보를 계속 전송하고, 여기서 복호화 과정이 추가로 계속되고, 그 다음 동글에 의해 컴퓨터로 다시 정보를 전송하는 것이고, 상기 엔티티에서 프로그램 코드는 결과적으로 실행될 수 있다.

본 발명은 컴퓨터 프로그램 코드의 암호화를 개략적으로 도시하는 도 3과 함께 컴퓨터 프로그램 요소의 적어도 일부의 암호화의 흐름도를 나타내는 도 1을 참조함으로써 설명될 것이다. 이런 암호화는 상기된 두 개의 엔티티들과 다른 제 3 엔티티 내에서 통상적으로 수행된다.

프로그램 코드를 암호화하기 위하여, 적어도 일부의 정적 데이터(306)는 본래 프로그램 코드(302)로부터 추출된다(단계 102). 따라서 여기서 프로그램 코드(304)로 표시된 추출된 정적 데이터(306)가 없는 본래 프로그램 코드(302)인 상기 본래 프로그램 코드(302)의 나머지들은 생성된다. 이 실시예에서, 본래 프로그램의 정적 데이터는 예를 들어, 문자열들, 설명들, 초기 가변 값들, 이미지들, 상수들, 포맷 관련 정적 데이터 또는 다른 정적 리소스들인 임의의 종류일 수 있다.

정적 데이터(306)를 추출한 후, 쌍 형태의 암호화/복호화 키들(공개 키 Kpb 314, 및 개인 키 Kpr 316)의 제 2 키는 생성된다(단계 104). 당업자에게 잘 알려진 바와 같이, 두 개의 키들 중 어느 하나는 데이터를 암호화하기 위하여 사용될 수 있고, 유사하게 다른 키는 데이터를 복호화하기 위하여 사용될 수 있지만, 하나의 키는 다른 하나가 상기 암호화된 데이터를 복호화하기 위하여 사용될 수 있을 때만 데이터를 암호화하기 위하여 선택된다.

여기서, 정적 데이터(306)는 암호화 키로서 공개 키 Kpb(314)를 사용하여 암호화되고(단계 106), 상기 공개 키(정적 데이터)Kpb(310)로 암호화된 정적 데이터를 생성한다. 상기된 통신 채널을 제공하기 위하여, 제 1 엔티티, 컴퓨터 및 제 2 엔티티, 동글 사이에서, 프로그램 코드(304)는 변형된 프로그램 코드(308)를 달성하기 위하여 변화된다(단계 108). 따라서, 이런 통신 채널은 하기될 데이터의 복호화 동안 사용될 것이다.

본 발명의 이 실시예에 따라, 프로그램 코드의 특정 위치에서 추출된(단계 102) 정적 데이터의 각 부분은 상기 데이터의 암호화된 카피에 의해 대체된다. 이것은 필수적으로 비암호화된 데이터가 본래 프로그램 코드(102)에 제공된 위치들에서가 아닌 본래 프로그램 코드에서 암호화된 데이터를 저장함으로써 수행된다. 프로그램 코드에 암호화된 정적 데이터가 저장된 후, 공개 키 Kpb(314)는 프로그램 코드에 저장되어(단계 112) 보호되는 프로그램 코드(312)를 얻는다(단계 116).

상기 공개 키 Kpb(314)에 해당하는 개인 키 KpR(316)는 동글(318)에 저장된다.

얻어진 보호된 프로그램 코드(312)는 따라서 암호화된 정적 데이터의 부분들을 포함하고, 상기 암호화된 정적 데이터는 상기 정적 데이터를 복호화하지 않고 프로그램 코드가 실행되는 것을 효과적으로 방지한다.

프로그램 코드 요소들의 특정 부분들, 즉 프로그램의 실행에 중요한 것이 암호화될 필요가 있다는 것은 명확하다. 이것은 모든 정적 데이터가 컴퓨터 프로그램 코드의 전체 부분들을 기능화하는 것을 방지하기 위하여 암호화될 필요가 없다는 것을 의미한다.

이와 같이 컴퓨터 프로그램 코드의 부분들을 복호화함으로써, 컴퓨터 프로그램 코드는 단일 if-then 설명들만을 해독함으로써 실행될 수 없다. 이것은 쉘형(shell like) 암호화 방법들과 대조되고, 상기 프로그램 코드는 큰 범위까지 암호화되지 않고 남겨지지만 상기 프로그램 코드의 실행을 방지하는 쉘이 암호화된다. 하나의 쉘을 해독함으로써, 쉘 내의 프로그램의 실행은 인에이블된다.

다음에는 암호화된 프로그램 코드의 실행중 암호화된 컴퓨터 프로그램 코드의 복호화를 기술할 것이다.

상기된 바와 같이, 동글에 대한 액세스없이 인증되지 않은 파티에 의한 프로그램 코드의 실행을 방지하기 위하여, 중요한 프로그램 코드 요소들의 암호화는 충분할 것이다. 프로그램 코드의 키 부분들을 실행할 능력없이, 프로그램 코드의 기능은 구현되지 않고, 적어도 완전히 구현되지 않는다.

중요 부분들만이 암호화되기 때문에, 암호화되지 않은 부분들은 실행될 수 있다. 프로그램 코드를 실행하는 중, 하기되는 방법은 프로그램 코드 요소의 각각의 부분에 사용된다. 각각의 부분에 대해, 통신 세션은 시작되고 정보는 컴퓨터 및 동글 사이의 통신 채널을 통하여 통신된다. 게다가, 각각의 상기 세션 동안, 세션 키는 하기에 보다 상세히 설명될 바와 같이 생성된다.

다음에서, 컴퓨터 프로그램 코드를 실행하는 중 정적 데이터를 복호화하는 보다 상세한 설명은 도 2A, 2B, 4 및 5를 참조하여 설명된다.

본 발명의 이 실시예에 따라, 보호된 컴퓨터 프로그램 코드(402)의 실행을 수행하는 것은 컴퓨터에서 시작된다. 보호된 컴퓨터 프로그램 코드 내에 컴퓨터는 도 3에서 공개 키 Kpb(314)로 암호화된 정적 데이터(static data)Kpb(406)를 배치한다. 또한, 컴퓨터는 보호된 컴퓨터 프로그램 코드에서 저장된 공개 키 Kpb(408)를 검색한다.

암호화된 정적 데이터를 마주치면, 랜덤 세션 키 Ks(404) 형태의 제 3 키는 생성된다(단계 202).

암호화된 정적 데이터(406)는 생성된 랜덤 세션 키 Ks(404)와 결합되고(단계 204), 상기 암호화된 정적 데이터(406) 및 세션 키 Ks(404)의 결합은 공개 키(406)를 사용하여 암호화되고(단계 206), 암호화된 정적 데이터(406), 및 상기 세션 키 Ks(404)의 암호화된 결합((static data)Kpb+ Ks)Kpb(410)을 생성한다.

이런 암호화된 결합(410)의 생성 후, 상기 암호화된 결합(410)은 동글(54)쪽으로 보내진다(단계 208). 도 4의 수직 점선(A)은 컴퓨터(52)와 동글(54) 사이의 인터페이스를 나타낸다.

동글은 컴퓨터의 일부 또는 예를 들어 인터넷인 임의의 종류의 네트워크를 통하여 접속을 사용함으로써 컴퓨터에 접속될 수 있다.

그 후, 단계(210)에서 컴퓨터(52)는 도 3의 공개 키 Kpb(314)로부터 복호화되지만, 랜덤 세션 키(Ks)(426)로 암호화된 정적 데이터(static data)Ks(430)를 동글(54=412)로부터 수신한다. 그 다음 컴퓨터는 세션 키 Ks(432)를 사용하여 암호화된 정적 데이터를 복호화한다(212).

랜덤 세션 키가 대칭 키이기 때문에, 암호화 및 복호화는 동일한 키를 사용하여 수행된다. 이것은 랜덤 세션 키(426), 세션 키(432), 및 랜덤 세션 키 Ks(404)가 동일한 키들이라는 것을 의미한다.

복호화후, 정적 데이터(434)는 얻어지고(단계 214), 상기 정적 데이터(434)는 프로그램 코드(436)의 실행 동안 요구시 사용된다.

상기는 암호화된 정적 데이터를 복호화하는 컴퓨터 내에서의 방법을 기술했다. 이하는 암호화된 정적 데이터를 복호화하는 동글에서의 방법을 기술한다.

본 발명의 이 실시예에 따라, 동글(54)은 우선 정적 데이터를 암호화하는 방법 동안 도 3에서 개인 키 Kpr(316)를 얻는다(단계 216). 둘째로, 1) 공개 키로 암호화된 정적 데이터(406), 및 2) 세션 키 Ks(404)의 암호화된 결합((정적 데이터) Kpb+ Ks)Kpb(410)를 수신하고(단계 218), 여기서 상기 결합은 공개 키 Kpb(408)로 암호화된다. 동글(54)로부터, 상기 암호화된 결합((static data)Kpb+ Ks)Kpb(414), 및 개인 키 Kpr(416)는 추출된다. 지금, 개인 키(416)를 사용함으로써, 암호화된 결합(414)은 복호화되고(단계 220), 세션 키 Ks(420), 및 공개 키 Kpb(408)로 암호화된 정적 데이터(static data)Kpb(418)를 생성한다. 이 복호화 다음, 암호화된 정적 데이터(418)는 개인 키(416)로 불리는 동일한 키인 개인 키 Kpr(422)를 사용하여 다시 복호화된다(단계 222). 이런 복호화후(단계 222), 복호화된 정적 데이터(424)는 얻어진다(단계 224).

따라서 동글이 복호화된 정적 데이터를 가진다. 복호화된 정적 데이터(424)는 다시 암호화되지만(단계 226), 이 단계에서 세션 키(426)를 사용하여, 상기 키는 암호화된 결합을 복호화하여 얻어진다(단계 220).

따라서, 도 3의 공개 키 Kpb(314)를 사용하여 수행되지만, 세션 키(426)를 사용하여 암호화된 초기 암호화로부터 복호화된 정적 데이터가 얻어진다. 이런 암호화된 정적 데이터(static data)Ks(428)는 동글(54)로부터 컴퓨터(52)로, 도 4의 B로 표시된 동글 컴퓨터 인터페이스를 통하여 보내진다(단계 228).

본 발명의 일 실시예에 따라, 이런 인터페이스(B)는 동글 컴퓨터 USB 인터페이스이다. 이 인터페이스는 인터넷 같은 네트워크, 다른 네트워크, 하나 이상의 다른 컴퓨터들, 또는 임의의 형태의 통신 채널을 포함하는 대안이다.

도 6은 그 위에 저장된 컴퓨터 프로그램 코드 수단을 가진 컴퓨터 프로그램 제품(62)을 도시한다. 이 컴퓨터 프로그램 제품은 임의의 형태, 예를 들어 컴팩트 디스크(CD), 디스켓, 디지털 다기능 디스크(DVD), 고체 메모리, 또는 하드 디스크일 수 있다.

컴퓨터 프로그램 코드의 실행 보호는 상기 컴퓨터 프로그램 코드에 따라 또는 몇몇 방식으로 제어되는 임의의 하드웨어에 인증되지 않은 액세스를 방지하기 위하여 사용될 수 있다. 적당한 능동 엔티티를 사용하여, 적당한 동글은 상기 하드웨어에 대한 액세스를 인증한다.

본 발명은 하기된 바와 같이 많은 방식으로 가변될 수 있다.

상기된 실시예에 대한 한가지 대안은 제 2 엔티티로서 보안 칩을 이용하는 것이다. 따라서, 비록 하나가 다른 하나 내부에 배치될지라도, 보안 칩 및 상기 컴퓨터가 두 개의 별개의 엔티티들이라는 것이 이해된다. 보안 칩들을 포함하는 보안 플랫폼의 한가지 예는 TCPA/팔라듐 플랫폼이고, 상기 플랫폼은 이런 다른 실시예에 사용되는데 적합하다.

본 발명의 다른 실시예에서 컴퓨터 프로그램 코드의 실행 보호는 다른 종류의 컴퓨터 프로그램 코드의 능동 엔티티를 사용하여 이루어질 수 있다. 따라서 이것은 보안 칩 또는 동글이 암호화된 정적 리소스들의 복호화에 사용되는 실시예들에 대한 대안이다.

다른 실시예에서 정적 데이터를 암호화하는 방법의 단계들의 순서는 변화되고 몇몇 단계들은 본 발명의 보호 범위와 다르지 않게 삭제될 수 있다. 예를 들어, 프로그램 코드를 변화시키는 단계(108)는 공개 및 개인 키들을 생성하는 단계전에 수행될 수 있다(단계 104).

본 발명의 다른 실시예에서, 정적 데이터를 복호화하는 방법은 컴퓨터에 의해 암호화된 데이터를 동글에 전송하는 단계를 포함하고, 여기서 데이터는 개인 키를 사용하여 복호화되고 컴퓨터에 추가로 리턴된다. 이 실시예에서, 세션 키는 사용되지 않는다.

본 발명의 다른 실시예에서, 정적 데이터를 복호화하는 방법은 암호화된 데이터 및 세션 키를 컴퓨터에 의해 동글로 전송하는 단계를 포함한다. 동글은 정적 데이터를 복호화하고, 세션 키를 사용하여 정적 데이터를 암호화하고 상기 데이터를 컴퓨터로 리턴한다. 이 실시예는 세션 키 및 암호화된 정적 데이터의 결합을 암호화하기 위하여 공개 키를 사용하지 않는다.

본 발명의 다른 실시예에서, 정적 데이터를 복호화하는 방법에서 세션 키 및 암호화된 정적 데이터는 공개 키를 사용하여 독립적으로 암호화된다. 따라서, 세션 키 및 암호화된 정적 데이터의 암호화된 결합이 없다.

상기된 바와 다른 실시예에서, 세션 키는 컴퓨터에 의해서만 암호화되고, 반면 이미 암호화된 정적 데이터는 이전과 같이 동글로 전송된다.

다른 실시예에서, 컴퓨터 프로그램 복호화 디바이스는 몇몇 컴퓨터들을 포함하는 분산된 컴퓨터 디바이스이다.

본 발명의 다른 실시예에서, 컴퓨터 프로그램 요소의 특정 위치에서 추출된 정적 데이터는 동일하거나 다른 컴퓨터 프로그램 요소의 다른 위치에 저장된다. 암호화되지 않은 정적 데이터는 요소로부터 추출되고 그 위치에서 더 이상 사용할 수 없다.

다른 실시예에서, 암호화된 정적 데이터를 복호화하는 동안 세션의 생성은 프로그램 코드로부터의 주문후 컴퓨터에 의해 수행된다.

다른 실시예에서, 암호화된 정적 데이터를 복호화하는 동안 세션 키의 생성은 암호화된 정적 데이터의 새로운 부분을 마주하기 전에 프로그램 코드에 의해 수행된다.

다른 실시예에서, 제 1 엔티티는 PDA(퍼스널 디지털 어시스탄트), 팜 탑 컴퓨터, 랩탑 컴퓨터, 퍼스널 컴퓨터, 게임 컴퓨터, 컴퓨터 서버, 또는 유사한 것 같은 임의의 종류의 컴퓨터이다.

상기된 실시예들이 본 발명을 제한하기보다 오히려 도시하고, 당업자가 첨부된 청구항들의 범위에서 벗어나지 않고 많은 다른 실시예들을 설계할 수 있을 것이라는 것이 주목되어야 한다.

청구항들에서, 괄호 사이에 배치된 임의의 참조 부호들은 청구항을 제한하는 것으로 해석되지 않는다. 단어 "포함하다"은 청구항에 기술된 것과 다른 요소들 또는 단계들의 존재를 배제하지 않는다. 단어 요소 앞의 단수표현은 다수의 요소들의 존재를 배제하지 않는다. 본 발명은 몇몇 구별되는 요소들을 포함하는 하드웨어, 및 적당히 프로그램된 컴퓨터에 의해 실행될 수 있다.

몇몇 수단을 열거하는 디바이스 청구항에서, 이들 몇몇 수단은 하나 및 동일한 아이템의 하드웨어로 구현될 수 있다. 특정 방법들이 다르게 종속항들을 인용한다는 사실은 이들 방법들의 결합이 바람직하게 사용될 수 없는 것을 가리키지 않는다.

도면의 간단한 설명

도 1은 본 발명의 바람직한 실시예에 따라 암호화하는 방법의 흐름도.

도 2A는 컴퓨터 프로그램 코드를 가진 디바이스에서 수행되는, 본 발명의 바람직한 실시예에 따라 복호화하는 방법의 흐름도.

도 2B는 본 발명의 바람직한 실시예에 따라 복호화하는 방법의 흐름도.

도 3은 본 발명에 따른 프로그램 코드의 암호화를 개략적으로 도시한 도면.

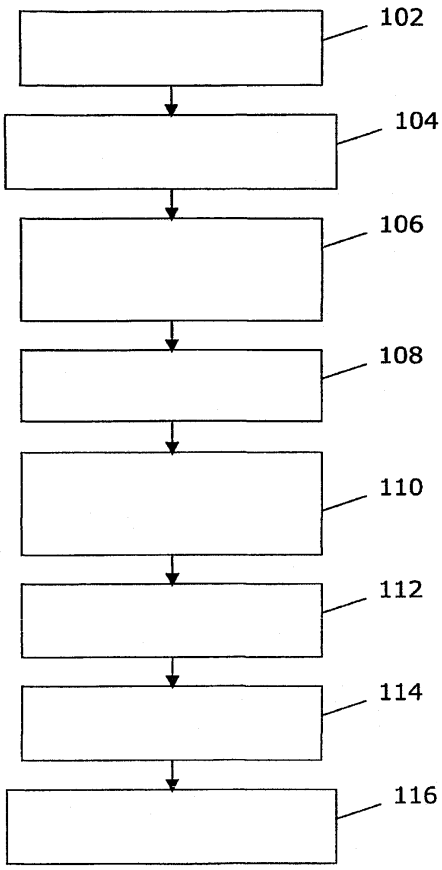
도 4는 본 발명에 따른 보호된 프로그램 코드의 복호화를 개략적으로 도시한 도면.

도 5는 암호화된 데이터를 복호화하는 동안 두 개의 엔티티들이 통신하는 컴퓨터 및 동글을 개략적으로 도시한 도면.

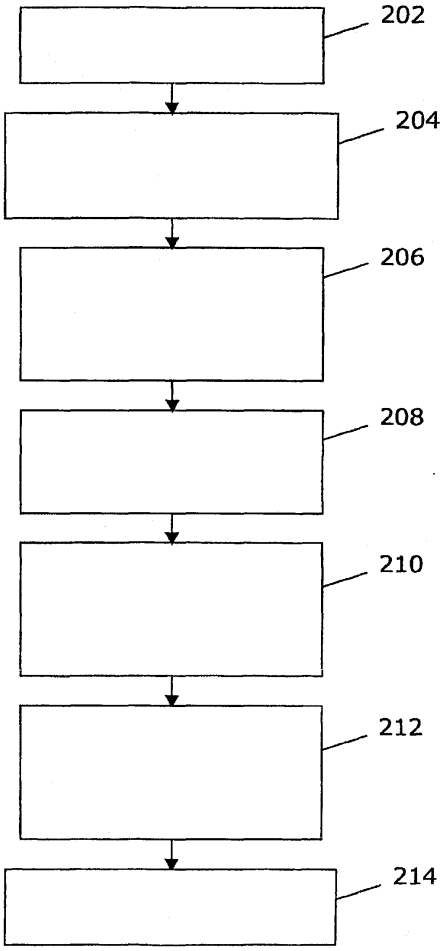
도 6은 본 발명과 관련된 그 위에 컴퓨터 프로그램 코드 수단을 가진 컴퓨터 프로그램 제품을 도시한 도면.

도면

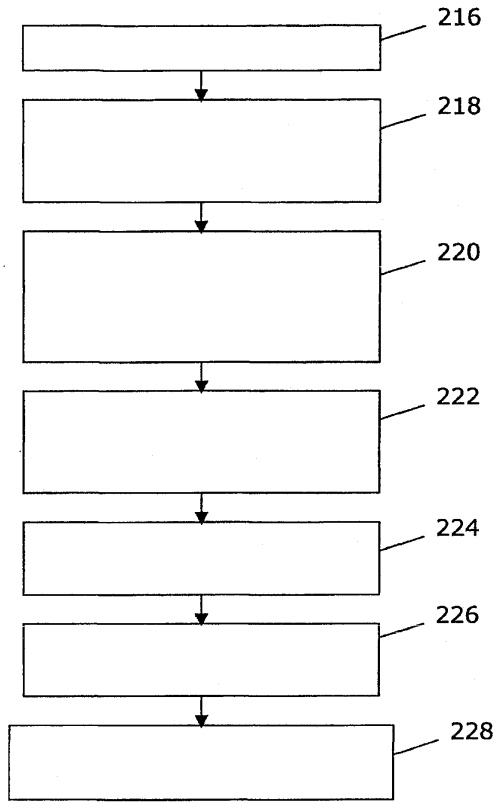
도면1



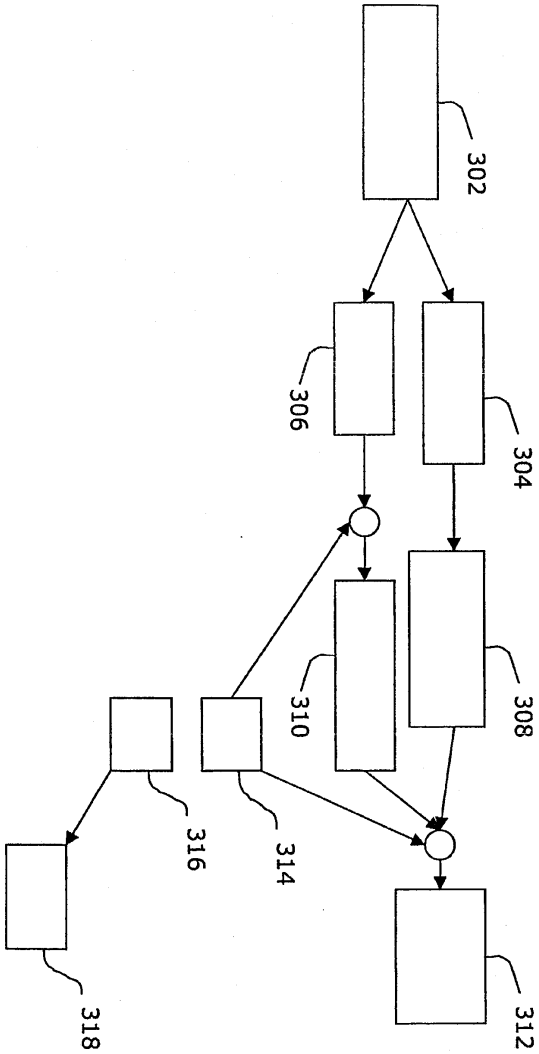
도면2A



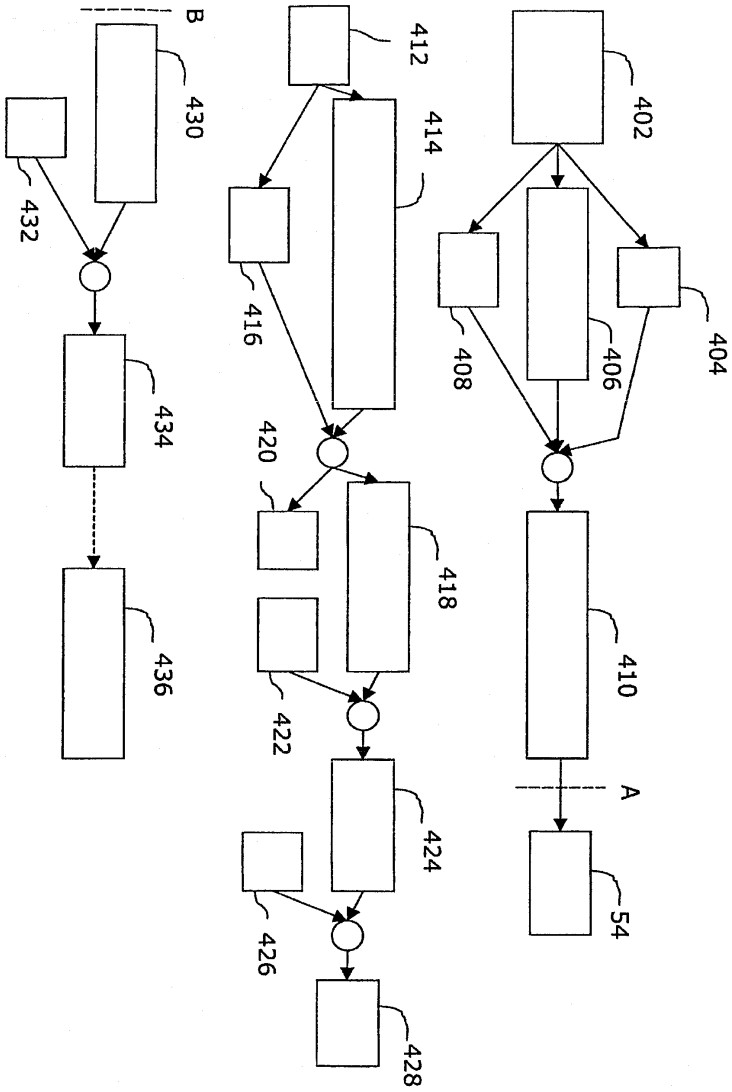
도면2B



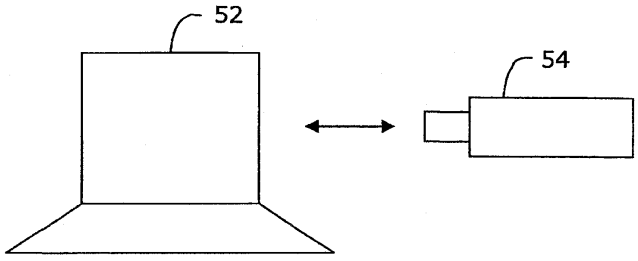
도면3



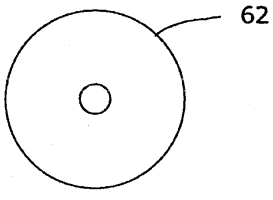
도면4


B
430
432
434
436
412
414
416
420
418
422
424
426
428
402
404
406
408
410
A
54

도면5


52
54

도면6


62