(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2012-0046508
(43) 공개일자 2012년05월10일

(51) 국제특허분류(Int. Cl.)
G06Q 40/00 (2006.01) *H04L 9/32* (2006.01)
H04L 9/28 (2006.01)
(21) 출원번호 10-2010-0108196
(22) 출원일자 2010년11월02일
심사청구일자 2010년11월02일

(71) 출원인
인하대학교 산학협력단
인천광역시 남구 인하로 100, 인하대학교 (용현동)
(72) 발명자
양대헌
인천광역시 남구 소성로 71, 하이테크센터 317호 (용현동, 인하대학교)
맹영재
인천광역시 남구 소성로 71, 하이테크센터 307호 (용현동, 인하대학교)
(74) 대리인
이원희

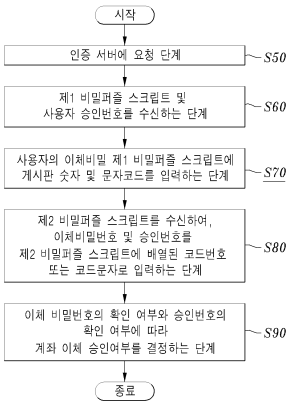
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **다양한 비밀퍼즐을 이용한 승인 방법**

(57) 요 약

다양한 비밀퍼즐을 이용한 승인 방법을 개시한다. 상기 다양한 비밀퍼즐을 이용한 승인방법은 계좌 이체에 따른 사용자 인증을 인증 서버에 요청하는 제1단계, 상기 인증 서버로부터 j×k 행렬로 이루어진 제1 비밀퍼즐 스크립트 및 사용자 승인 번호를 사용자 단말기에 수신하는 제2단계, 상기 비밀퍼즐 스크립트에 사용자의 이체비밀에 상응하는 숫자 또는 문자를 상기 비밀퍼즐 스크립트에 제시된 코드 번호 또는 코드 문자로 입력하고, 사용자의 계좌 이체비밀번호를 상기 제1 비밀퍼즐 스크립트에 제시된 코드 번호로 입력하는 제3단계, 상기 인증 서버로부터 j×k 행렬로 이루어진 제2 비밀퍼즐 스크립트를 수신하여, 상기 승인번호에 상응하는 상기 제2 비밀퍼즐 스크립트에 배열된 코드 번호 또는 코드 문자로 입력하는 제4단계, 상기 이체 비밀번호의 확인 여부와 상기 승인 번호의 확인 여부 따라 계좌 이체 승인 여부를 결정하는 제5단계를 포함한다.

대 표 도 - 도4



이 발명을 지원한 국가연구개발사업

과제고유번호	41473-01
부처명	교육과학기술부
연구사업명	기본연구
연구과제명	인지기반 온라인 금융 서비스와 DDoS 대응 시스템
주관기관	인하대학교
연구기간	2010.05.01 ~ 2013.04.30

특허청구의 범위

청구항 1

계좌 이체에 따른 사용자 인증을 인증 서버에 요청하는 제1단계;

상기 인증 서버로부터 j×k 행렬로 이루어진 제1 비밀퍼즐 스크립트 및 사용자 승인 번호를 사용자 단말기에 수신하는 제2단계;

상기 비밀퍼즐 스크립트에 사용자의 이체비밀에 상응하는 숫자 또는 문자를 상기 비밀퍼즐 스크립트에 제시된 코드 번호 또는 코드 문자로 입력하고, 사용자의 계좌 이체비밀번호를 상기 제1 비밀퍼즐 스크립트에 제시된 코드 번호로 입력하는 제3단계;

상기 인증 서버로부터 j×k 행렬로 이루어진 제2 비밀퍼즐 스크립트를 수신하여, 상기 승인번호에 상응하는 상기 제2 비밀퍼즐 스크립트에 배열된 코드 번호 또는 코드 문자로 입력하는 제4단계; 및

상기 이체 비밀번호의 확인 여부와 상기 승인 번호의 확인 여부 따라 계좌 이체 승인 여부를 결정하는 제5단계를 포함하는 다양한 비밀 퍼즐을 이용한 승인 방법.

청구항 2

상기 제3단계는,

사용자 이체 비밀의 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 a)단계;

상기 사용자의 이체 비밀번호의 첫번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 주변기기등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 b)단계;

사용자 이체 비밀의 두 번째 문자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 c)단계;

상기 사용자의 이체 비밀번호의 두 번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열된 숫자를 주변기기 등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 d)단계; 및

상기 이체 비밀 및 상기 이체 비밀번호의 자리수와 동일 수만큼 상기 a) 및 d) 단계를 반복하는 e)단계를 포함하는 비밀퍼즐을 이용한 승인 방법.

청구항 3

제1항에 있어서,

상기 제4단계는,

사용자 이체 비밀 번호의 첫 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 f)단계;

인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 g)단계;

사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 h)단계;

인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열

된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 i)단계; 및

상기 이체 비밀 번호의 자리수와 동일하게 상기 f) 및 i) 단계를 반복하는 j)단계를 포함하는 비밀퍼즐을 이용한 승인 방법.

청구항 4

제2항에 있어서,

상기 g)단계는,

상기 제1 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 하는 단계인 것을 특징으로 하는 비밀퍼즐을 이용한 승인 방법.

청구항 5

제2항에 있어서,

상기 i)단계는,

상기 비밀 퍼즐 스크립트의 제4행에 배열된 숫자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 수행하는 단계인 것을 특징으로 하는 비밀퍼즐을 이용한 승인 방법.

청구항 6

제3항에 있어서,

상기 g)단계는,

상기 제2 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 하는 단계인 것을 특징으로 하는 비밀퍼즐을 이용한 승인 방법.

청구항 7

제3항에 있어서,

상기 i)단계는,

상기 제2 비밀 퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 수행하는 단계인 것을 특징으로 하는 비밀퍼즐을 이용한 승인 방법.

청구항 8

제3항에 있어서,

상기 g)단계와 상기 i)단계는,

상기 제1 비밀 퍼즐 스크립트 상에서 배열된 상기 이체 비밀 번호 문자에 상응하는 숫자를 선택하여 서버에서 지시한 방향으로 승인 번호의 숫자를 움직이도록 수행하는 것을 특징으로 다양한 비밀퍼즐을 이용한 승인 방법.

청구항 9

제1항에 있어서,

상기 이체 비밀은,

상기 사용자의 수신자명, 전화번호, 휴대폰 번호, 주소, 회사, 나이, 수신자 이미지로 나타낼 수 있는 것을 특징으로 하는 다양한 비밀퍼즐을 이용한 승인 방법.

청구항 10

제1항에 있어서,

상기 제1 비밀퍼즐 스크립트 및 제2 비밀퍼즐 스크립트는,

보안 등급에 따라 추가 더미 스크립트를 포함하는 것을 특징으로 하는 다양한 비밀퍼즐을 이용한 승인 방법.

명 세 서

기 술 분 야

[0001] 본 발명은 암호화 방법 및 이를 이용한 시스템에 관한 것으로, 보다 상세하게는 인증 서버로부터 암호화 또는 해시 함수의 결과값을 수신 및 기반으로 하여 비밀퍼즐을 생성하여 다양한 비밀 퍼즐을 이용한 승인 방법에 관한 것이다.

배 경 기 술

[0002] 인터넷뱅킹은 우리 생활에 없어서는 안 될 매우 중요한 전자금융거래 수단으로 자리 잡았다. 국내의 인터넷뱅킹은 2010년도 2/4분기 중 일평균 3,291만건, 29조 9,548억원으로 꾸준한 증가세를 유지하고 있고, 전체 입출금 및 자금이체의 34.1%, 조회서비스의 66.1%로 금융서비스 전달채널 중 가장 높은 비중을 차지하고 있다.

[0003] 특히 모바일뱅킹은 일평균 262만건, 4,078억원으로 전분기대비 각각 13.2%, 14.0%의 증가율을 기록하는 등 앞으로도 더욱 활성화 될 전망이다.

[0004] 인터넷뱅킹은 비대면 거래인만큼 사용자 인증과 거래승인 과정을 보호하는 것이 매우 중요하다. 이를 위해 인터넷뱅킹에는 다양한 비밀들이 사용되고 있으나, 그 이용방법은 입력장치를 통해 평문을 그대로 입력하는 형태로 모두 동일하다.

[0005] 평문으로 입력되는 비밀 코드는 수학적으로 보안성이 검증된 인증 프로토콜과 암호기법을 통해 보호받기 때문에 네트워크 수준에서 일어날 수 있는 대다수의 공격으로부터는 안전하나, 비밀 코드 없이 암호를 해독하는 것은 막대한 비용과 시간이 요구된다는 사실과 비밀의 단순한 이용방법을 알고 있는 공격자는 자연스럽게 비교적 공격하기 쉬운 비밀 코드 입력과정을 목표로 하는 것이다.

[0006] 그러한 공격의 대표적인 예가 사용자가 입력하는 단계에서 비밀을 얻어내는 키보드 후킹(Keyboard Hooking)과 사용자의 눈을 속여 공격자가 제작한 사이트에 비밀을 입력하도록 유도하는 피싱(Phishing 또는 파밍:Pharming, 이하 피싱)공격이다. 그리고 한층 더 지능화된 공격으로 문서변조가 있다.

[0007] 문서변조는 사용자의 거래가 시작되는 시점부터 완료되기까지 사용자의 눈을 속여 공격자의 의도대로 거래를 완료시키는 공격이다. 비밀 코드를 단순히 얻어내는 것이 목적인 키보드 후킹이나 피싱과 다르게 입력된 비밀 코드를 사용자의 의심 없이 악용하기 때문에 더욱 지능화된 공격이다.

[0008] 공격자는 자신의 문서 위에 문서원본의 내용부분만이 겹치도록 만들고 이렇게 변조된 문서를 사용자에게 보여준다. 문서의 내용에 이상이 없음을 확인한 사용자는 문서에 사인을 한다.

[0009] 사인된 문서를 받은 공격자는 겹친 부분을 떼어내어 사용자가 사인한 위조된 문서를 가지게 된다. 사용자는 자신이 거래하고자 하는 내용을 눈으로 확인하고 비밀 코드를 입력하기 때문에 사용자의 눈을 속일 수 있으면 사용자의 의심 없이 비밀 코드를 얻어낼 수 있는 것이다.

[0010] 이 공격은 단순히 비밀 코드를 증명하는 방법이 문서변조 공격 앞에서는 의미가 없다는 것을 보여준다.

[0011] 인터넷뱅킹에는 다양한 형태의 비밀 코드가 사용되고 있다. 현재 사용되고 있는 비밀 코드는 대부분이 고정된 형태의 비밀 코드로, 주로 사용자를 인증하기 위해 사용되고 있다. 사용자를 인증하기 위해 사용되는 비밀 코드는 사용자가 외우는 것, 지니는 것, 신체의 일부로 나눌 수 있으며 인터넷뱅킹에는 주로 외우는 형태와 지니는 형태의 비밀 코드를 혼합하여 사용하고 있다(Multi-factor authentication).

[0012] 현재 사용 중인 비밀 코드들을 아래에 기재된 [표 1]에 정리하였다.

표 1

[0013]

대분류	소분류	비밀의 종류	비고
고정된 비밀코드	외우는 형태의 비밀 코드	로그인 패스워드	영문,숫자 혼합 6~10자리 서버에는 암호화되어 저장되어 있을 수 있음
		공인인증서패스워드	암호화되어있는 공인인증서를 사용하기 위한 패스워드, 서버는 이 패스워드를 가지고 있지 않음
		계좌(거래)비밀번호	계좌이체 등을 위한 패스워드 서버에는 암호화되어 저장되어 있을 수 있음
	파일 형태의 비밀 코드	공인인증서	하드디스크, 이동식저장장치, 휴대폰 등에 저장될 수 있음 외우는 형태의 비밀과 함께 사용됨.
		ISP (Internet Secure Payment)	
	소유하는 형태의 비밀 코드	보안카드	비밀의 수가 제한됨(예,4자리 숫자 35개) 질의-응답형태로 사용됨 서버는 그 응답을 평문형태로 알 수 있음
유동적인 비밀코드	소유하는 형태의 비밀 코드	OTP (One Time Password)	매번 다른 6자리 숫자를 생성해냄. 복사가 불가능함 서버는 OTP의 응답을 평문형태로 알 수 있음

[0014] 다양한 형태의 비밀 코드(예컨대, 고정된 비밀 코드 또는 유동적인 비밀 코드)는 용도에 따라서는 사용자 인증을 위한 것과 거래 승인을 위한 것, 공격자의 입장에서는 고정된 형태와 유동적인 형태로 나누어 볼 수 있다.

[0015] 사용자 인증과정에는 고정된 형태의 비밀 코드만이 사용되고 있는데, 고정된 형태의 비밀 코드는 한번 노출되면 사용자가 해당 비밀 코드를 바꾸기 전까지 공격자가 악용할 수 있다는 단점이 있다. 그 때문에 사용자 인증만으로 할 수 있는 것은 조회서비스로 한정되고, 그 외의 중요정보 조회나 정보변경, 거래승인 등에서는 소유하는

형태의 비밀을 이용하여 매번 다른 응답(이하 승인응답)을 증명하도록 되어있다. 매번 달라지는 응답은 분명 공격자에게 부담이 되는 것이다.

[0016] 하지만, 소유하는 형태의 비밀 코드를 포함한 모든 비밀 코드는 평문을 그대로 입력하도록 되어있다. 이렇게 여전히 그 이용방법이 단순하다. 단순한 비밀 코드 이용방법의 단점은 그 비밀 코드를 입력하는 과정에서 평문이 노출될 수 있다는 점, 거래 내역과 연관되어있지 않아 입력된 비밀 코드가 사용자의 의도와는 다르게 사용될 수 있다는 것이다.

[0017] 이는 매번 응답이 변하도록 되어있는 승인응답이라도 예외가 아니다. 응답이 유동적이라 하더라도 그 응답이 어느 용도로 사용되는 것인지 확인할 수 없다는 것이 문제이다.

[0018] 자동화된 악성코드(이하 악성코드): 사용자의 컴퓨터(또는 휴대폰, 이하 컴퓨터)에 설치되는 악성코드는 해당 기기의 모든 권한을 가진다고 가정한다. 악성코드는 사용자의 비밀 코드를 얻어내기 위해 사용자의 입력을 훔쳐보는 키보드 후킹이나 공격자가 제작해놓은 사이트에 비밀을 입력하도록 유도하는 피싱, 거래를 공격자의 의도대로 완료되도록 문서를 변조하는 문서변조 공격 등이 가능하다.

[0019] 악성코드에 대응하기 위한 보안프로그램은 동일한 권한을 가지는 악성코드에 대한 근본적인 해결책이 되지는 못한다.

[0020] 하지만 사람과 로봇을 구분하기 위해 인지능력을 이용하는 CAPTCHA(Completely Automated Public Turing test to tell Computers and Humans Apart)는 풀 수 없다.

[0021] 공격자의 실시간 참여(이하 원격의 공격자): 악성코드의 능력에 더해서 악성코드가 원격에 위치하는 공격자와 실시간으로 통신이 가능하여 사용자의 화면을 볼 수 있다면 단순히 사람과 로봇을 구분하기 위한 CAPTCHA는 무의미하다. 더해서, 공격자는 사용자가 위치한 곳에 카메라를 설치하여 사용자가 키보드에 입력하는 비밀뿐만 아니라, 사용자가 소유하는 형태의 비밀 코드 또한 모두 훔쳐볼 수 있다고 가정한다(어깨너머 훔쳐보기: Shoulder-Surfing Attack).

[0022] 일반적으로, 악성코드와 원격의 공격자의 차이점은 공격과정에 공격자(사람)가 직접 실시간으로 참여하였는지 여부이다. 공격자가 악성코드에 감염된 다수의 컴퓨터를 제어할 수 있다면 이들을 동시에 동원하여 대규모의 공격을 수행할 수 있다.

[0023] 이 공격은 DDoS(분산 서비스 거부 공격: Distributed Denial-Of-Service attack)를 의미하기도 하지만, 사용자에게 금전적인 피해를 직접적으로 입히는 동시다발적인 문서변조 공격을 의미하기도 한다. 자동화된 공격이 불가능하도록 CAPTCHA가 적용된 시스템에는 원격의 공격자가 이에 대응할 수 있다.

[0024] 하지만 공격과정에 사람이 직접 참여해야 하기 때문에 자동화된 악성코드와 같은 대규모 공격에는 한계가 있고 실시간으로 통신해야 공격이 가능하다는 점은 공격자의 위치를 파악할 수 있는 단서가 될 수 있기 때문에 공격자의 공격을 제한시킬 수 있다.

[0025] 문서변조는 사용자의 컴퓨터에 거래가 발생할 때 사용자의 눈을 속여 공격자의 의도대로 거래를 완료시키는 지능화된 공격이다. 이 공격은 악성코드가 웹 브라우저에 상주하는 MITB(Man In The Browser)를 통해 가능하며, 공격자가 문서변조 공격을 자동화하기 위해서는 다음과 같은 과정을 거쳐야한다.

[0026] 첫째, 소스코드의 분석이 필요하다.

[0027] 동적인 웹 문서를 구성하는데 필수 요소인 스크립트는 인터프리터가 소스코드를 직접 해석한다. 국내 전자금융거래의 경우 ActiveX를 통해 중요한 내용(스크립트 포함)을 암호화하기도 하지만 복호화가 스크립트를 통해 이루어지기 때문에 DebugBar와 같은 프로그램을 이용, 암호화된 문서에서 복호화 함수를 직접 호출하여 소스코드를 얻어낸 후 소스코드를 분석할 수 있었다.

[0028] 둘째, 공격대상 문서의 탐지가 필요하다.

[0029] 일반적으로 서버에서 문서를 구분하는 방법이 URL(Uniform Resource Locator)이라는 것을 고려하면, 악성코드는 단순히 URL을 읽어 공격 대상을 구분해낼 수 있다. 예로, onDocumentComplete이벤트를 이용하여 다운로드가 완료된 문서의 URL를 얻어낼 수 있었다.

[0030] 셋째, 스크립의 삽입이 필요하다.

[0031] 악성코드가 삽입될 수 있는 경로는 TLS/SSL이 적용된 문서의 경우 전송(Transport)계층 이후의 구간에서 스크립트를 삽입할 수 있고 플러그인을 통한 부분적인 복호화가 요구되는 경우에는 인터넷 익스플로러(Internet Explorer)의 유저 인터페이스(User Interface)까지 모든 구간에서 스크립트 삽입이 가능하다.

[0032] 넷째, 공격자가 의도하는 데이터의 입력이 필요하다.

[0033] 키보드 드라이버의 인터럽트 핸들러가 호출하는 keybd_event(Winuser.h에 정의되어 있음)를 이용하면 키보드 보안 프로그램에 공격자가 원하는 데이터를 입력할 수 있었다.

[0034] 스크립트의 삽입이 가능할 때 문서를 변조하는 방법은 두 가지이다.

[0035] 가짜 문서를 제작하여 CSS의 z-index속성을 이용, 문서 원본에 계층구조 형태로 덮어씌우는 방법과 스크립트의 함수 오버라이딩(Function Overriding)을 통해 서버와 통신하는 데이터만을 변조하는 방법이다.

[0036] 키보드 보안 프로그램은 국내에서 쉽게 찾아볼 수 있는 비밀 보호방법이다. 이 프로그램이 하는 일은 두 가지로, 키보드 후킹을 막아 사용자 비밀의 노출을 방지하는 것과 사용자의 키 입력을 키보드 드라이버 수준에서 암호화한 것을 응용계층에 전달하여 비정상적인 키 입력 데이터를 방지하는 것이다.

[0037] 하지만 실험결과 이 두 보안기법 모두 근본적으로 악성코드에 대응하지 못하고 있다는 것이 밝혀졌다. 악성코드는 사용자가 입력하는 비밀 코드를 얻어내기 위해서 가짜 입력 폼을 이용할 수 있다. 비밀 입력 폼을 암호화대상이 아닌 것으로 바꾸고 비밀이 입력되면 그 폼에 DOM(Document Object Model)으로 접근하여 비밀을 훔쳐가는 것이다.

[0038] 공격자는 비밀 코드를 훔쳐가기 위해서 보안기능이 최소화된(또는 Class ID만 같고 동작은 안하는) 키보드 보안 프로그램[9,10]을 악용할 수도 있다. 이 프로그램은 사용자가 동의하는 경우에 제공되는데, 이 프로그램이 설치되어 있는 경우에는 키보드 후킹에 안전하지 못하게 된다. 공격자는 사용자의 컴퓨터에 보안기능이 최소화된 키보드 보안 프로그램을 강제로 설치하고 키보드 후킹을 시도하면 비밀을 얻어낼 수 있는 것이다.

[0039] 악의적인 키 입력과 관련하여, 공격자는 윈도우에서 제공하는 화상 키보드나 원격 데스크톱으로도 인터넷 뱅킹이 가능하다는 사실에서 키보드 보안 프로그램이 적어도 인터럽트 처리 수준에서 동작하지 않는다는 것을 알 수 있다. 이러한 키보드 보안 프로그램은 인터럽트 후킹 수준의 키로거를 막을 수 없고, 키보드 보안 프로그램이 정상적으로 동작하는 도중에도 윈도우 API수준의 keybd_event로 암호화 대상인 계좌비밀번호와 계좌번호를 입력하는 것이 가능했다. 실험해보지 못한 다른 키보드 보안 프로그램에서도 같은 결과를 보일 것이라 예상한다.

[0040] 가상키보드는 키보드 후킹으로부터 사용자의 비밀을 보호하기 위해 비밀입력을 물리적인 키보드가 아닌 가상의 키보드 상에서 마우스를 통해 입력하도록 하는 방법이다. 물리적인 키보드의 키 입력 데이터가 여러 구간을 통과하는 것과는 다르게, 가상키보드는 응용계층에서 독립적으로 동작하도록 설계되어있기 때문에 비밀노출에 대한 위협이 응용계층으로 한정된다.

[0041] 이는 비교적 안전해보일 수 있지만, 응용계층에 상주하는 악성코드에게는 여전히 비밀이 노출될 수 있다. 예로, 국내 ㅁ은행에 최근 도입된 가상키보드는 매번 키의 위치가 바뀌도록 되어있으며, 사용자가 이를 이용하고자 할 때 가상키보드 이미지를 다운로드하고 화면에 출력한다. 이때 키의 위치정보가 담긴 XML(Extensible Markup Language)파일 또한 다운로드하고 이를 가상키보드 이미지에 적용한다.

[0042] 사용자가 마우스를 이용해 가상이미지상의 비밀을 클릭하면 스크립트는 이벤트의 좌표를 암호화하여 서버에 전송하도록 되어있다. 실험결과 악성코드는 DOM으로 가상키보드 이미지에 접근하거나 다운로드할 수 있었으며, transkey1.pressKey함수를 오버라이딩하여 사용자가 마우스로 입력한 x, y좌표를 얻어낼 수 있었다. 가상키보드 이미지와 사용자가 입력한 좌표만 있다면 악성코드가 간단한 이미지 처리기법과 문자인식을 통해 비밀을 얻어내거나, 원격에 위치한 공격자가 직접 이미지상에 좌표를 대입하여 비밀을 얻어낼 수 있는 것이다.

[0043] 가상키보드에서 비밀을 얻어내는 또 다른 방법으로, 가상키보드 이미지를 서버가 전송한 것이 아닌 자신이 준비해놓은 가상키보드 이미지를 대신 출력하는 방법도 있다.

[0044] 이 경우, 악성코드는 준비해놓은 가상키보드의 키 좌표정보를 사전에 알고 있기 때문에 사용자가 입력한 좌표에 해당하는 비밀을 해석하기 위해 이미지 처리기법을 사용하지 않아도 된다. 다른 가상키보드 모두 같은 공격에 취약할 것으로 예상된다.

[0045] 사용자가 가진 비밀 코드가 노출되기 쉽다는 이유를 비밀 코드의 entropy가 낮다는 점에 착안하여, 문자로 이루어진 비밀 코드 대신 entropy가 비교적 높은 이미지를 비밀로 사용하는 연구가 활발히 진행되고 있다. 인증화면이 나타나면 맨 왼쪽 위부터 시작하여 사용자가 알고 있는 이미지를 따라 이동하다가 가장자리에 도달하면 그에 해당하는 위치의 숫자를 이용하여 인증하는 것이다.

[0046] 이러한 기법은 여러 개의 이미지를 외워야 한다는 부담이 있고 인증에 소요되는 시간이 길며(한 라운드에 15~20초) 에서 안전성 또한 지적되어 실용성이 낮다고 평가된다.

[0047] 이와 같이 이미지를 이용한 기법들은 높은 엔트로피(entropy)를 제대로 활용하지 못하거나, 어깨너머 훔쳐보기에 취약하다는 공통적인 단점이 있다. 문자 비밀코드의 경우 키보드를 통해 비밀 코드 그대로를 컴퓨터에 입력할 수 있지만 이미지를 키보드와 마우스를 이용하여 컴퓨터에 입력하는 것은 쉬운 일이 아니다.

[0048] 그렇기 때문에 서버에서 전송한 이미지들 즉, 비밀 코드의 후보군 중에서 선택하도록 되어있는데 제한되어 있는 크기의 해상도는 출력되는 비밀 코드의 후보군 개수 또한 제한되어 그만큼 보안성이 떨어지게 된다.

[0049] 비밀 코드의 후보군이 많거나 또는 여러 라운드를 반복하게 하는 것은 사용자의 인식시간이 길어지거나 부담을 주기 쉽기 때문에 사용성이 떨어지게 된다. 사용성이 떨어지기도 하지만 공격자는 어깨너머 훔쳐보기로 사용자의 성공적인 인증시도를 바라보면 비밀을 얻어낼 수 있다. 이와 같은 취약점을 극복하기 위해 비밀을 직접 선택하는 것이 아닌, 비밀의 predicate를 외우도록 하고 인증과정에서 이에 추가적인 연산을 통해 응답을 하도록 하였다.

[0050] 하지만 이러한 방법은 사용자에게 인지능력에 더하여 머릿속으로 연산능력을 추가로 요구하기 때문에 사용성이 떨어지고 사용자가 정할 수 있는 predicate가 한정적이라는 연구는 그러한 접근의 보안성 또한 높지 않다는 것을 뒷받침 한다.

[0051] 따라서 더 높은 entropy를 가지는 비밀과 관련한 연구는 그 입력방법과 관련하여, 사용성과 보안성 모두를 충족할 수 있는 연구가 진행중이다.

[0052] 현재 사용되는 문자(alphanumeric)로 이루어진 비밀 코드의 호환성을 고려하여, 비밀 코드 증명 방법만을 바꾸어 안전하게 인증하는 방법 또한 연구되고 있다. S3PAS의 경우에는 pass-icon대신 문자비밀코드를 이용하도록 하였다.

[0053] pass-icon은 화면에 보이는 아이콘들 중에서 자신이 외운 아이콘 세 개를 꼭짓점으로 가지는 삼각형을 머릿속으로 그리고 그 삼각형 안의 아이콘을 선택하도록 되어있다. 아이콘이 임의적인 순서로 나타나는 pass-icon과는 다르게 S3PAS는 사용자가 외우는 자리의 패스워드를 번째 인증 단계에서 세 자리씩 잘라 사용하였다.

[0054] 앞과 뒤 라운드에서 두 자리의 패스워드가 동시에 사용된다는 점은, 사용자가 선택한 문자를 중심으로 삼각형을 재구성하는 패스워드 후보군의 추측 가능성을 높이게 된다. 공격자가 이 패스워드 후보군을 여러 개 모으면 교차공격(패스워드 후보군들의 교집합)을 통해 패스워드를 알아낼 수 있게 된다. 또한 머릿속에서 패스워드를 세 자리씩 기억해내는 것은 사용자에게 부담이 될 수 있다.

[0055] 또 다른 연구에서는 두 겹으로 전달되는 6x6행렬(알파벳 26개, 숫자 10개로 이루어져있으며 초기단계에서는 두 겹의 행렬에 표시된 문자가 같다)을 서버가 전송하는 질의로 사용하고, 이 행렬상의 움직임을 통해 인증하는 방법을 소개하였다.

[0056] 인증은 총 n(패스워드의 길이)단계이며 사용자는 홀수 번째 인증단계에서 임시 비밀을 행렬 상에서 임의적으로 선택하고 짝수 번째 인증단계에서는 이전(홀수 번째) 인증단계에서 선택한 임시 비밀을 이용하도록 되어있다. 번째 인증단계에서 사용자는 행렬 상에서 번째 패스워드가 위치한 행과 번째 패스워드가 위치한 열이 겹치는 교차점을 찾고, 이 교차점(상위계층)을 임시비밀(하위계층)에 방향키로 이동하여 위치시킨다.

[0057] 이 기법은 무차별 공격과 재전송 공격, 교차공격을 적절히 방어하지만 머릿속에서 패스워드를 두 자리씩 기억하고 임시비밀코드 또한 기억해야 한다는 점은 사용자에게 부담이 될 수 있다.

[0058] 또 다른 연구에서는 사용자에게 안전한 채널을 통해 질의전달이 가능한 경우, 첫 번째 패스워드 문자와 같은 열에 위치하는 pass-object를 임시비밀코드로 사용하는 경우, 안전한 채널을 통해 pass-object를 전달받는 경우에 대한 인증기법을 보였다. 첫 번째와 세 번째 기법의 경우는 질의를 전달할 수 있는 안전한 채널이 요구되는데, 이 논문에서 언급한 음성을 통한 방법 또한 악성코드가 해석할 수 있기 때문에 안전한 채널이라고 납득하기 어렵다.

[0059] 만약 질의를 안전하게 전달하기 위해 부가장치를 이용하도록 가정한다면, 동일한 가정에서 부가장치를 이용한 다른 편리한 기법을 사용할 수 있기 때문에 효율적이지 않게 된다.

[0060] 피싱의 취약점은 사용자가 접근하고자 하는 서버에 존재하는 것이 아니라 사용자의 컴퓨터에 감염되어 있는 악성코드, 사용자의 URL입력 실수 또는 공격자가 공존하는 취약한 네트워크 환경에서 발생하기 때문에 일반적인 보안방법으로는 대응하기가 쉽지 않다.

[0061] 그 예로, 어떤 홈페이지에 접근할 때만 실행되는 안티피싱 프로그램은 사용자가 URL을 실수로 입력했을 때 또는 악성코드가 HTTP request을 수정하여 피싱사이트로 유도한 경우에는 그 프로그램이 실행되지도 않기 때문에 해결책이 될 수 없다. 사용자 컴퓨터의 메모리에 계속 상주하여 사용자가 접근하는 URL 감시방법은 주로 화이트리스트 및 블랙리스트를 이용하는데, 리스트를 이용한 방법은 알려지지 않은 공격에 대해서는 여전히 무방비이다.

[0062] 더해서, 이 리스트들은 주로 URL을 검사하고 사용자가 보는 화면이 사용자가 접근하고자 하는 홈페이지의 화면인지는 확인하지 않는다. 피싱방지 프로그램이 화면캡쳐 등을 통해 실제화면을 비교한다 하여도 공격자는 이를 쉽게 우회할 수 있다.

[0063] 공격자는 실제화면과 동일한 화면을 직접 제작할 필요 없이 정상적인 서버로부터 홈페이지 내용을 다운로드 받아다가 사용자에게 그대로 전달하되 비밀을 훔치는 코드만을 추가할 수 있는 것이다. 이러한 피싱사이트는 사용자에게 서비스를 정상적으로 제공하기 때문에 사용자가 의심을 가지기도 쉽지 않다.

[0064] 사용자가 화면이 전혀 다른 피싱사이트로 접근했다 하여도 해당 사이트가 개편되었다는 문구를 보인다면 이 경우 또한 사용자는 의심을 가지지 않을 수 있다.

[0065] CAPTCHA를 이용한 부정승인 방지방법은 거래내역을 포함시킨 형태의 확장된 CAPTCHA는 자동화된 악성코드에 대응할 수 있지만 구현방법에 따라 악성코드에 취약할 수 있다. ArcotVPS의 경우 배경과 문자열의 색이 다르고 이

미지에서 문자가 차지하는 비율이 낮다는 사실을 근거로 Kmeans++를 사용하면 문자열을 분리시킬 수 있다.

[0066] 악성 프로그램이 캡차를 읽을 수 없더라도, 추출된 4개의 문자열 중 하나를 정해 추출한 다음 변조된 거래내역에 대한 승인이미지를 생성할 수 있는 것이다. 따라서 악성 프로그램은 CAPTCHA를 읽을 수 없어도 1/4의 확률로 공격에 성공할 수 있다.

[0067] MS워터마크는 일반폰트로 거래내역이 표현된 배경 위로 CAPTCHA가 표현되어 있다.

[0068] 이 기법에서는 CAPTCHA의 폰트가 더욱 두껍고 크기 때문에 윤곽선 검출(Edge Detection)의 Canny 필터(가우시안 마스크를 통해 잡영 제거 후 Sobel필터 적용)를 통해 강한윤곽선만을 남긴 후에 이를 추출해내면 CAPTCHA 문자열을 분리시킬 수 있다.

[0069] 특정 두께에 대한 영역을 세선화(Thinning) 하여 문자열을 얻어낸 다음이 이를 다시 Blur 필터 등으로 두껍게 만들면 더욱 선명한 문자열을 얻어낼 수 있다. 분리된 캡차 이미지는 변조된 거래내역에서 악용될 수 있다.

[0070] 또한, 위의 두 기법은 CAPTCHA를 인식하고 그에 대한 응답을 사용자가 수동적으로 입력해야 하도록 되어있다.

[0071] 문맥 기반 CAPTCHA에서는 사용자에게 전용되는 CAPTCHA이미지 중에서 사용자의 거래와 관련된 것만을 선택하도록 하였다. 이 기법에서 6개의 콘텐츠(수신은행, 수신계좌, 수신자명, 이체금액, 송신자, 송신은행)와 44개의 더미 캡차로 구성되어 있을 경우, 무차별 선택에 대한 보안성은 이다. 공격이 가능한(사용자가 입력한 수신은행과 이체금액은 그대로 사용) 최소한의 콘텐츠 변경은 수신계좌와 수신자명 두 가지를 변경하는 것으로, 이때의 보안성이 높을 수 있다.

[0072] 이 기법에서는 사용자가 여러 개의 CAPTCHA를 인식해야 하기 때문에 편의성이 높지는 않다. 하지만 사용자가 CAPTCHA를 읽고 입력하는 형태가 아닌 선택하는 용도로 사용하였기 때문에 더욱 복잡도가 높은 CAPTCHA를 이용할 수 있다. 예로, 심하게 뒤틀려 일부 영역이 인식하기 힘든 형태가 되어도 CAPTCHA를 선택하는 데에는 지장이 없다.

[0073] 부가장치를 이용한 승인방법은 부가장치에서 거래내역을 출력하고 이 장치를 통해 승인하도록 하는 것이 목적이다. 현재 사용 중인 부가장치를 이용한 승인방법으로는 인터넷뱅킹 계좌이체의 전화승인서비스가 유일하다. 전화승인서비스는 계좌이체 승인단계에서 ARS 시스템을 이용해 사용자에게 전화를 걸어 거래내역을 확인시켜 주고 거래를 승인하거나 취소할 수 있도록 하는 방법으로 사용자에게 편의성을 제공하지 못한다.

[0074] 트랜잭션 서명 기법은 사용자 토큰이라 불리는 추가적으로 발급받은 기기에 거래내역의 일부를 입력하면 토큰에 출력된 MAC을 사용자가 컴퓨터에 수작업으로 입력하는 방법이다. 상기 트랜잭션 서명 기법은 사용자 토큰에 인증서를 저장한 형태 또는 사용자가 입력한 MAC을 컴퓨터에서 공인인증서를 통해 사인하는 방법으로 부인방지를 추가적으로 제공한다. 트랜잭션 서명 기법은 사용자 입장에서는 일부이기는 하지만 거래내역을 한 번 더 입력해야 하기 때문에 편의성이 떨어진다.

[0075] IBM의 ZTIC은 USB 장치의 디스플레이를 통해 거래내역을 출력하고 두 개의 버튼을 통해 승인 또는 취소를 결정하도록 한다[30]. ZTIC은 컴퓨터에 연결되면 USB 저장장치로 인식이 됨과 동시에 사전에 정의되어있는 사이트(은행)으로 연결되도록 하는 프록시를 설정하고 웹 브라우저와 해당 사이트와의 통신은 모두 ZTIC을 통해 이루어지며 이 세션을 암호화하는 TLS/SSL의 키는 악성 프로그램이 접근할 수 없도록 되어있다. 내용을 확인하고 버튼으로 승인여부를 결정하면 되기 때문에 편의성이 좋지만 추가적인 장치를 발급해야 한다는 단점이 있다.

[0076] 부가장치를 통해 승인을 하는 방법은 보안성은 높지만 거래 당시에 추가적인 장치를 소지하고 있어야 승인이 가능하기 때문에 이동성이 떨어진다고 할 수 있다.

발명의 내용

해결하려는 과제

[0077] 본 발명이 해결하고자 하는 과제는 다양한 비밀퍼즐을 이용하여 키보드 후킹이나 피싱공격으로부터 사용자의 비밀이 노출되는 것을 사전에 방지하고, 사용자 승인이 요구되는 거래 정보가 공격자에 의해 변조되는 것을 방지할 수 있는 다양한 비밀 퍼즐을 이용한 승인 방법을 제공하는 것이다.

과제의 해결 수단

[0078] 상기 과제를 해결하기 위한 본 발명의 다양한 비밀퍼즐을 이용한 승인 방법은 계좌 이체에 따른 사용자 인증을 인증 서버에 요청하는 제1단계, 상기 인증 서버로부터 j×k 행렬로 이루어진 제1 비밀퍼즐 스크립트 및 사용자 승인 번호를 사용자 단말기에 수신하는 제2단계, 상기 비밀퍼즐 스크립트에 사용자의 이체비밀에 상응하는 숫자 또는 문자를 상기 비밀퍼즐 스크립트에 제시된 코드 번호 또는 코드 문자로 입력하고, 사용자의 계좌 이체비밀번호를 상기 제1 비밀퍼즐 스크립트에 제시된 코드 번호로 입력하는 제3단계, 상기 인증 서버로부터 j×k 행렬로 이루어진 제2 비밀퍼즐 스크립트를 수신하여, 상기 승인번호에 상응하는 상기 제2 비밀퍼즐 스크립트에 배열된 코드 번호 또는 코드 문자로 입력하는 제4단계, 상기 이체 비밀번호의 확인 여부와 상기 승인 번호의 확인 여부 따라 계좌 이체 승인 여부를 결정하는 제5단계를 포함한다.

[0079] 상기 제3단계는 사용자 이체 비밀의 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 a)단계, 상기 사용자의 이체 비밀번호의 첫번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 주변기기등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 b)단계, 사용자 이체 비밀의 두 번째 문자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 c)단계, 상기 사용자의 이체 비밀번호의 두번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열된 숫자를 주변기기등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 d)단계 및 상기 이체 비밀 및 상기 이체 비밀번호의 자리수와 동일 수 만큼 상기 a) 및 d) 단계를 반복하는 e)단계를 포함한다.

[0080] 상기 제4단계는 사용자 이체 비밀 번호의 첫 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 f)단계, 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 g)단계, 사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 h)단계, 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 i)단계 및 상기 이체 비밀 번호의 자리수와 동일하게 상기 a) 및 d) 단계를 반복하는 j)단계를 포함한다.

[0081] 상기 g)단계는 상기 제1 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 하는 단계인 것을 특징으로 한다.

[0082] 상기 i)단계는 상기 비밀 퍼즐 스크립트의 제4행에 배열된 숫자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 수행하는 단계인 것을 특징으로 한다.

[0083] 상기 g)단계는 상기 제2 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 하는 단계인 것을 특징으로 한다.

[0084] 상기 i)단계는 상기 제2 비밀 퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 수행하는 단계인 것을 특징으로 한다.

[0085] 상기 g)단계와 상기 i)단계는 상기 제1 비밀 퍼즐 스크립트 상에서 배열된 상기 이체 비밀 번호 문자에 상응하는 숫자를 선택하여 서버에서 지시한 방향으로 승인 번호의 숫자를 움직이도록 수행하는 것을 특징으로 한다.

[0086] 상기 이체 비밀은 상기 사용자의 수신자명, 전화번호, 휴대폰 번호, 주소, 회사, 나이, 수신자 이미지로 나타낼 수 있는 것을 특징으로 한다.

[0087] 상기 제1 비밀퍼즐 스크립트 및 제2 비밀퍼즐 스크립트는 보안 등급에 따라 추가 더미 스크립트를 포함하는 것을 특징으로 한다.

발명의 효과

[0088] 본 발명에 따른 비밀 퍼즐을 이용하여 승인 여부를 실시할 경우, 사용자가 피싱사이트에 접근했다 하여도 비밀은 노출되지 않는 효과가 있다.

[0089] 또한, 사용자는 비밀퍼즐을 통해 비밀을 입력하도록 되어있다는 사실을 알기 때문에 비밀을 직접 입력하는 폼은 이용하지 않을 수 있고, 피싱 사이트에서 비밀퍼즐을 적용하여도 공격자는 이체비밀번호와 승인응답을 알아내지 못하는 효과가 있다.

[0090] 또한, 본 발명은 이체비밀이 포함된 비밀퍼즐을 통과해야 계좌이체가 이루어지기 때문에 수신계좌정보를 변조하려는 악성코드와 사용자의 컴퓨터를 실시간으로 제어하는 공격자로부터 안전하게 거래를 승인할 수 있다.

[0091] 또한, 본 발명의 비밀 퍼즐을 이용한 승인방법은 높은 해상도를 요구하지 않고 사용법이 간편하기 때문에 모바일 뱅킹에도 적용될 수 있다.

도면의 간단한 설명

[0092] 도 1은 본 발명의 일 실시 예에 따른 비밀 퍼즐을 이용한 승인 방법을 나타낸 플로우 챠트이다.

도 2는 도 1의 제3단계를 보다 구체적으로 나타낸 플로우 챠트이다.

도 3은 도 1의 승인 방법에 적용된 비밀퍼즐을 나타낸 예시도이다.

도 4는 본 발명의 다른 실시 예를 나타낸 다양한 비밀퍼즐을 이용한 승인 방법을 나타낸 플로우 챠트이다.

도 5는 도 4에 기재된 제3단계를 보다 구체적으로 나타낸 플로우 챠트이다.

도 6은 도 4에 기재된 제4단계를 보다 구체적으로 나타낸 플로우 챠트이다.

도 7a는 본 발명의 승인 방법에 적용된 다양한 비밀퍼즐의 예를 나타낸 예시도이다.

도 7b는 본 발명의 승인 방법에 적용된 다양한 비밀퍼즐의 다른 예를 나타낸 예시도이다.

도 7c는 본 발명의 승인 방법에 적용된 다양한 비밀퍼즐의 다른 예를 나타낸 예시도이다.

도 7d는 본 발명의 승인 방법에 적용된 다양한 비밀퍼즐의 다른 예를 나타낸 예시도이다.

발명을 실시하기 위한 구체적인 내용

[0093] 아래에서는 첨부한 도면을 참고로 하여 본 발명의 실시 예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 여기에서 설명하는 실시 예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0094] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 "~부","~기" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0095] 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시 예를 예시하는 첨부 도면 및 도면에 기재된 내용을 참조하여야 한다.

[0096] 본 발명을 설명함에 앞서, 현재 사용중인 승인응답(보안카드 또는 OTP: One Time Password)이 가지는 공통점은 숫자로 이루어져있다는 점과 매번 다른 응답을 사용하도록 되어있다는 점, 그리고 서버가 그 응답의 평문을 알 수 있다는 점이다. 보안카드의 경우 비밀의 수가 제한되어있기는 하지만 2자리씩 두번, 질의-응답 형태로 사용되기 때문에 보안카드의 비밀 모두 또는 그 비밀들을 복원시킬 수 있는 인자를 서버가 저장하고 있다.

[0097] OTP의 경우에는 질의-응답 방식, 시간 동기화 방식, 이벤트(카운터) 동기화 방식, 조합(시간 및 카운터) 방식이 있으며, 이들 방식 모두 응답을 생성하기 위해 필요한 인자가 인증 서버와 공유 또는 동기화되어 있기 때문에 서버가 응답을 평문상태로 알 수 있다. 이러한 승인 응답이 가지는 속성은 질의(질의-응답 프로토콜)로 사용하기에 적합하다.

[0098] 이와는 다르게 이체비밀번호(또는 계좌비밀번호, 이하 이체비밀번호)는 고정된 형태의 비밀로, 인증 서버에 암호화 일방향 해시 함수의 결과 값이 저장되어 있을 수 있기 때문에 서버가 평문을 알고 있지 않을 수도 있다.

[0099] 따라서, 아래에서 기술할 본 발명은 이 두 가지 비밀이 가지는 속성을 바탕으로 구성하도록 한다.

[0100] 도 1은 본 발명의 일 실시 예에 따른 비밀 퍼즐을 이용한 승인 방법을 나타낸 플로우 챠트이며, 도 2는 도 1의 제3단계를 보다 구체적으로 나타낸 플로우 챠트이다.

[0101] 도 1에 도시된 바와같이, 본 발명의 비밀 퍼즐을 이용한 승인 방법은 제1단계(S10) 내지 제4단계(S40)를 포함한다.

[0102] 상기 제1단계(S10)는 계좌 이체에 따른 사용자 인증을 인증 서버에 요청하는 단계일 수 있다.

[0103] 상기 제2단계(S20)는 상기 인증 서버로부터 생성된 비밀퍼즐 스크립트 및 사용자 승인 번호를 사용자 단말기에 수신하는 한계일 수 있다.

[0104] 상기 제3단계(S30)는 사용자의 이체 비밀번호 및 승인 번호에 상응하는 숫자 또는 문자를 상기 비밀퍼즐 스크립트에 표시된 코드 번호 또는 코드 문자로 각각 입력하는 단계일 수 있다.

[0105] 상기 제4단계(S40)는 상기 이체 비밀번호의 확인 여부에 따라 상기 승인번호를 확인한 후, 승인 여부를 판단하

는 단계일 수 있다.

[0106] 보다 구체적으로, 상기 제1단계(S10)는 사용자가 계좌 이체를 위해 사용자 컴퓨터 또는 휴대단말기를 통해 인증 서버로 계좌이체 승인 요청을 수행하는 단계일 수 있다.

[0107] 상기 컴퓨터 또는 휴대단말기는 상기 인증 서버와 유?무선 접속이 가능한, 휴대용단말기, 휴대인터넷장치(Mobile Internet Device), MP3, MP4, PMP(Portable Multimedia Player), 노트북 컴퓨터, UMPC(Ultra Mobile PC), 또는 PC(personal computer), 휴대전화(cellular phone), 스마트폰, 또는 PDA(personal digital assistants)와 같은 각종 전자장치가 될 수 있다.

[0108] 또한, 휴대단말기일 경우, SIM(subscribe identity module)을 내장하는 기존의 CDMA(code division multiple access)용 단말기뿐만 아니라, 상기 SIM을 탑재한 카드를 탈부착할 수 있는 GSM(Global system for mobile communication)용 단말기를 포함할 수 있다.

[0109] 또한, 상기 컴퓨터 또는 휴대단말기 내부에는 상기 인증 서버와 접속이 가능한 응용 프로그램이 설치된다.

[0110] 상기 제2단계(S20)는 상기 인증 서버로부터 상기 컴퓨터 또는 휴대단말기에서 전송된 사용자 인증 요청에 상응하는 응답을 수신하는 단계로서, 인증 서버는 암호화 일 방향 해시 함수의 결과값을 상기 인증 서버와 연결된 데이타베이스에서 추출하여 비밀퍼즐 스크립트 변환된 데이타를 전송한다. 또한, 사용자 승인 번호를 사용자 단말기로 전송하는 단계일 수 있다.

[0111] 상기 제3단계(S30)는 도 2를 참조하면, a)단계(S31) 내지 e)단계(S35)를 포함할 수 있다.

[0112] 상기 a)단계(S31)는 사용자 이체 비밀 번호의 첫 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 단계일 수 있다.

[0113] 상기 b)단계(S32)는 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0114] 상기 c)단계(S33)는 사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 단계일 수 있다.

[0115] 상기 d)단계(S34)는 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0116] 상기 e)단계(S35)는 상기 이체 비밀 번호의 자리수와 동일하게 상기 a) 및 d) 단계를 반복하는 단계일 수 있다.

[0117] 여기서 상기 b)단계(S32)는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 선택하여 상기 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 하는 단계일 수 있다.

[0118] 상기 d)단계(S34)는 상기 비밀 퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3항의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 수행하는 단계일 수 있다.

[0119] 상기 b)단계(S32)와 상기 d)단계(S34)는 상기 비밀 퍼즐 스크립트 상에서 배열된 상기 승인 번호의 숫자에 상응

하는 숫자를 선택하여 서버에서 지시한 방향으로 승인 번호의 숫자를 움직이도록 수행하는 단계일 수 있다.

[0120] 도 3은 도 1의 승인 방법에 적용된 비밀퍼즐을 나타낸 예시도이다.

[0121] 예를 들어, 도 3을 참조하면, 회색바탕에 표현된 번호가 이체 비밀 번호(고정적임)이고 그 아래 나열된 번호는 승인 응답(유동적임: 키보드나 마우스를 통해 움직일 수 있음)이다.

[0122] i번째 자리의 이체비밀번호를 i번째의 회색 바탕 행에서 찾고, 그 아래에 i번째 자리의 승인응답을 i번째 화살표가 가리키는 방향으로 이동시킨다.

[0123] 구체적인 예로, 사용자의 이체비밀번호가 '3869'이고 승인응답이 '6989'라면 이때의 이동정보는 왼쪽 7칸, 왼쪽으로 1칸, 오른쪽으로 8칸, 오른쪽으로 0칸 이 된다.

[0124] 이체비밀번호 그리고 승인응답이 순차적으로 표현되어 있지만 이를 임의적으로 표현하거나, 캡차(CAPTCHA)형태로 표현할 수도 있다.

[0125] 인증 서버에서는 본 발명의 비밀퍼즐을 통해 이체비밀번호와 승인응답을 동시에 확인할 수 있기 때문에 계좌이체정보 입력단계(예컨대, b)단계)에서 이체비밀번호를 별도로 입력하지 않아도 된다.

[0126] 또한, 이체비밀번호와 승인응답의 역할을 바꾸어서 사용할 수도 있다. 하지만 고정된 곳에 위치시켜야 승인응답을 위치시킬 곳을 매번 새로 찾지 않아도 되어 사용자가 본 발명인 비밀퍼즐에 쉽게 익숙해질 수 있다.

[0127] 상기 캡차(CAPTCHA)는 상기 컴퓨터 사용자가 사람인지를 판단하기 위해 사용되는 일종의 테스트. 스팸 소프트웨어의 자동 계정 등록을 막기 위해 계정 등록할 때 거치는 테스트을 말한다.

[0128] 예를 들어, 단어를 시각적으로 변형시킨 복잡한 패턴을 배경 화면 위에 보여 줌으로써 메일 계정을 등록하려는 사용자는 해당 단어를 키보드로 입력해 계정을 생성할 수 있으며, 이와 같이 상기 캡차 패턴은 일부러 변형시킨 글자로 사람은 쉽게 알아볼 수 있지만 스팸 컴퓨터는 알아보지 못한다. 따라서 스패머는 자동으로 계정을 생성할 수 없게 되고, 스팸 메일 발송도 어려워지게 된다.

[0129] 도 4는 본 발명의 다른 실시 예를 나타낸 다양한 비밀퍼즐을 이용한 승인 방법을 나타낸 플로우 챠트이며, 도 5는 도 4에 기재된 제3단계를 보다 구체적으로 나타낸 플로우 챠트이며, 도 6은 도 4에 기재된 제4단계를 보다 구체적으로 나타낸 플로우 챠트이다.

[0130] 도 4에 도시된 바와 같이, 비밀퍼즐을 이용한 승인 방법은 제1단계(S50) 내지 제5단계(S90)를 포함한다.

[0131] 상기 제1단계(S50)는 계좌 이체에 따른 사용자 인증을 인증 서버에 요청하는 단계일 수 있다.

[0132] 상기 제2단계(S60)는 상기 인증 서버로부터 j×k 행렬로 이루어진 제1 비밀퍼즐 스크립트 및 사용자 승인 번호를 사용자 단말기에 수신하는 단계일 수 있다.

[0133] 상기 제3단계(S70)는 상기 비밀퍼즐 스크립트에 사용자의 이체비밀에 상응하는 숫자 또는 문자를 상기 비밀퍼즐 스크립트에 제시된 코드 번호 또는 코드 문자로 입력하고, 사용자의 계좌 이체비밀번호를 상기 제1 비밀퍼즐 스크립트에 제시된 코드 번호로 입력하는 단계일 수 있다.

[0134] 상기 제4단계(S80)는 상기 인증 서버로부터 j×k 행렬로 이루어진 제2 비밀퍼즐 스크립트를 수신하여, 상기 승인번호에 상응하는 상기 제2 비밀퍼즐 스크립트에 배열된 코드 번호 또는 코드 문자로 입력하는 단계일 수 있다.

[0135] 상기 제5단계(S90)는 상기 이체 비밀번호의 확인 여부와 상기 승인 번호의 확인 여부 따라 계좌 이체 승인 여부를 결정하는 단계일 수 있다.

[0136] 보다 구체적으로, 도 5를 참조하면, 상기 제3단계는 a)단계(S71) 내지 e)단계(S75)를 포함할 수 있다.

[0137] 상기 a)단계(S71)는 사용자 이체 비밀의 첫 문자에 상응하는 상기 비밀퍼즐 스크립트의 제1행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 단계일 수 있다.

[0138] 상기 b)단계(S72)는 상기 사용자의 이체 비밀번호의 첫번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 주변기기등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0139] 상기 c)단계(S73)는 사용자 이체 비밀의 두 번째 문자에 상응하는 상기 비밀퍼즐 스크립트의 제3행에 배열된 코드 번호 또는 코드 문자를 주변기기등을 통하여 클릭하는 단계일 수 있다.

[0140] 상기 d)단계(S74)는 상기 사용자의 이체 비밀번호의 두번째 숫자에 상응하는 상기 비밀퍼즐 스크립트의 제4행에 배열된 숫자를 주변기기등을 이용하여 상기 이체 비밀이 클릭된 코드 번호 또는 코드 문자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0141] 상기 e)단계(S75)는 상기 이체 비밀 및 상기 이체 비밀번호의 자리수와 동일 수 만큼 상기 a) 및 d) 단계를 반복하는 단계일 수 있다.

[0142] 상기 b)단계(72)는 상기 제1 비밀퍼즐 스크립트의 제2행에 배열된 숫자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 하는 단계일 수 있다.

[0143] 상기 d)단계(74)는 상기 비밀 퍼즐 스크립트의 제4행에 배열된 숫자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀이 클릭된 코드 문자 위치와 동일한 열로 움직이도록 수행하는 단계일 수 있다.

[0144] 다음으로, 도 6을 참조하면, 상기 제4단계(S80)는 보다 상세하게 a)단계(S81) 내지 e)단계(S85)를 포함할 수 있다.

[0145] 상기 f)단계(S81)는 사용자 이체 비밀 번호의 첫 숫자에 상응하는 제2 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 단계일 수 있다.

[0146] 상기 g)단계(S82)는 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 제2 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0147] 상기 h)단계(S83)는 사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 상기 제2 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 단계일 수 있다.

[0148] 상기 i)단계(S84)는 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 상기 제2 비밀퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계일 수 있다.

[0149] 상기 j)단계(S85)는 상기 이체 비밀 번호의 자리수와 동일하게 상기 a) 및 d) 단계를 반복하는 단계일 수 있다.

[0150] 상기 g)단계(S82)는 상기 제2 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 선택하여 상기 제2행의 양쪽 끝에 표시된 방향으로 움직여 상기 제1행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 하는 단계일 수 있다.

[0151] 상기 i)단계(S84)는 상기 제2 비밀 퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 선택하여 상기 제4행의 양쪽 끝에 표시된 방향으로 움직여 상기 제3행의 이체 비밀번호가 입력된 숫자의 위치와 동일한 열로 움직이도록 수행하는 단계일 수 있다.

[0152] 상기 g)단계(S82)와 상기 i)단계(S84)는 상기 제2 비밀 퍼즐 스크립트 상에서 배열된 상기 이체 비밀 번호 문자에 상응하는 숫자를 선택하여 서버에서 지시한 방향으로 승인 번호의 숫자를 움직이도록 수행하는 단계일 수 있다.

[0153] 여기서, 상기 이체 비밀은 상기 사용자의 수신자명, 전화번호, 휴대폰 번호, 주소, 회사, 나이, 수신자 이미지로 나타낼 수 있다. 그러나 이에 한정되지는 않는다.

[0154] 상기 제1 비밀퍼즐 스크립트 및 제2 비밀퍼즐 스크립트는 보안 등급에 따라 추가 더미 비밀퍼즐 스크립트(예컨대, 도 7b에 도시된 비밀퍼즐 스크립(30))를 포함할 수 있다.

[0155] 예컨대, 상기 제1 비밀퍼즐 또는 제2 비밀퍼즐 중 어느 하나는 아래에 기재된 도 7a 내지 도 7d에 나타낸 비밀퍼즐(20 내지 50)을 이용하여 나타낼 수 있다.

[0156] 도 7a 내지 도 7b는 다양한 비밀퍼즐의 예를 나타낸 예시도이다.

[0157] 도 7a는 내지 도 7d를 참조하면, 계좌이체와 관련된 정보 중에서, 인증 서버와 사용자는 알고 있지만 사용자가 입력하지는 않는 정보(이하, 이체비밀)가 있다면 이를 이용하여 이체승인을 위한 다양한 비밀퍼즐들을 만들 수 있다.

[0158] 도 7a에 기재된 바와같이, 비밀퍼즐(20)은 홀수 행 또는 짝수 행에 이체비밀을 나타낼 수 있으며, 그 다음 행에 이체 비밀번호를 입력할 수 있다.

[0159] 예를 들어, 이체비밀이 수신자명 예컨대, "최미재" 이고, 이체비밀번호가 "251" 라 가정할 경우, 상기 본 발명의 다른 실시 예의 제3단계에서 설명한 바와같이, 제1행, 제3행, 제5행에는 "최", "미", "재" 라는 문자 코드를 선택하며, 상기 제2행의 코드 번호 2를 오른쪽으로 2칸 이동시키며, 제4행의 코드 번호 5를 왼쪽으로 2칸 이동시키며, 제6행의 코드 번호 1을 오른쪽으로 4칸 이동시킴으로서 인증 확인을 달성할 수 있다.

[0160] 상기 이체 비밀의 예로, 수신자명을 나타내었지만, 다른 예로, 전화번호, 휴대폰 번호, 주소, 회사, 나이, 수신자, 이미지 등이 될 수 있다.

[0161] 도 7b는 이체비밀로 사용자의 수신자명만이 기재될 경우, 보안 등급을 높이기 위해 더미 스크립트가 추가된 예를 나타낸 비밀퍼즐 스크립트(30)일 수 있다.

[0162] 도 7c는 사용자의 편의성을 위한 비밀퍼즐로서, 예컨대, 필요에 따라 이체비밀의 자릿수에 맞춘 비밀 퍼즐 스크립트(40)을 나타낸 예이다.

[0163] 도 7d는 다수의 계좌 이체 정보를 이체할 경우 예컨대, 6건의 계좌이체를 처리할 경우에 수신될 수 있는 비밀퍼즐 스크립트(50)의 예를 나타낸다.

[0164] 예를 들어, 제1행, 제3행, 제5행, 제7행, 제9행, 제11행에는 6명의 수신자명이 나타날 수 있으며, 각 짝수 행은 인증 번호 또는 계좌이체비밀번호에 상응하는 코드 번호를 주변기기 등을 통하여 상기 6명의 수신자명과 동일한 열의 짝수 행으로 이동시킴으로써 인증 방법을 달성할 수 있다.

[0165] 이하는, 본 발명의 실시 예를 이용하여 사용자의 안전한 계좌 이체 방법을 간단하게 설명하도록 한다.

[0166] 우선, 사용자는 송신계좌번호, 이체금액, 수신자 은행, 수신자 계좌번호를 입력 또는 선택하고 인증 서버에 계좌이체를 요청한다(S100).

[0167] 다음으로, 인증서버는 계좌이체 가능여부를 확인하고 이체에 문제가 없는 경우 이체비밀번호를 이용한 비밀퍼즐과 부분승인응답 질의를 클라이언트에 전송한다. 사용자는 승인응답을 조회하고 부분승인응답을 이체비밀번호를 주변기기(예컨대, 키보드 또는 마우스)를 이용하여 상기 비밀퍼즐 내에 배열된 숫자를 원하는 위치로 이동시킨다(S110).

[0168] 만약, 계좌이체가 2건 이상인 경우에는 S100과 S110를 반복할 수 있다.

[0169] 다음단계로 넘어가면 공인인증서 창이 나타나며, 공인인증서 창의 상단에는 이체정보를 확인하도록 되어있을 수 있고 중간에는 이체비밀번호를 묻는 비밀퍼즐(송신계좌 수만큼), 하단에는 공인인증서 패스워드를 입력하도록 한다. 이때 이체정보에서 이체비밀과 관련된 내용은 표시하지 않고 대신에 S110에서 사용자에게 보인 비밀퍼즐을 보여주도록 한다(S120).

[0170] 사용자는 승인응답을 이체비밀번호에 위치시키고 공인인증서 패스워드를 입력하여 서버에 전송하면 계좌이체가 완료된다(S130). 이체비밀은 여전히 표시하지 않고 수신자명만을 CAPTCHA로 표시하도록 한다.

[0171] 여기서, 이체비밀은 여전히 표시하지 않고 수신자명만을 CAPTCHA로 표시하도록 한다.

[0172] 이론적으로 숫자로 이루어진 비밀은 자릿수가 인 경우 만큼의 보안성을 제공한다. 4자리의 이체비밀번호와 4자리의 승인응답을 함께 사용하는 경우 이론상으로는 만큼의 보안성을 제공한다. 숫자로 이루어진 비밀을 복합적으로 사용한 비밀퍼즐의 이론적인 보안성은 승인응답의 자릿수(또는 퍼즐의 개수)가 인 경우, 에 머물지만 원격의 공격자에게도 안전하다는 장점이 있다. 시간이 동기화된 OTP를 승인응답으로 사용하는 경우에는 사용자가 입력할 때의 시간차를 고려해야 하기 때문에 사전에 정의해놓은 허용범위 만큼의 오차를 허용할 수 있다. 이때의 이론적인 보안성은 가 된다.

[0173] 기존의 비밀증명은 비밀을 직접적으로 입력하는 방법을 이용한다. 그 때문에 비밀의 이론적인 보안성과는 관련없이 공격자는 사용자가 입력하는 비밀을 키로거, 피싱/파밍 등의 공격을 통해 얻어내고 공격자의 의도대로 사용할 수 있었다. OTP와 같은 유동적인 비밀은 공격자의 의도대로 사용될 수 있는 시간을 단축시킬 뿐, 문서변조 공격에 취약하기 때문에 근본적인 해결책이 되지 못한다.

[0174] 따라서, 본 발명에서 제안하는 비밀퍼즐이 적용되었을 경우, 사용자가 피싱사이트에 접근했다 하여도 비밀은 노출되지 않는다. 사용자는 비밀퍼즐을 통해 비밀을 입력하도록 되어있다는 사실을 알기 때문에 비밀을 직접 입력하는 폼은 이용하지 않을 것이고, 피싱사이트에서 비밀퍼즐을 적용하여도 공격자는 이체비밀번호와 승인응답을 알아내지 못한다.

[0175] 문서변조 공격은 거래내역이 반영되어있지 않은 비밀이 어느 용도로 사용되었는지를 확인할 장치가 마련되어있지 않기 때문에 발생한다. 비밀퍼즐2는 공격자가 모르는 정보에 승인응답을 위치하도록 되어있기 때문에, 공격자가 서버가 전송한 비밀 퍼즐을 자신의 거래를 완료시킬 수 있는 형태로 조작할 수 없다.

[0176] 웹의 스크립트 수준에서 구현될 수 있으면 브라우저 호환성이 좋고 설치비용이 낮다. 추가적인 소프트웨어의 설치가 요구되면 사용자가 사용 중인 특정 브라우저 또는 운영체제에 따라 설치가 불가할 수도 있기 때문에 호환성이 높지 않고, 설치비용 또한 높아진다. 부가장치를 이용한 방법의 경우 보안성이 높은 반면에 하드웨어 기기 발급에 따른 비용 부담이 있고 이 기기를 소지하고 있을 때만 거래가 가능하기 때문에 이동성이 낮다. 호환성의 경우 브라우저 호환성과 현재 사용되고 있는 비밀 호환성을 동시에 평가하였다.

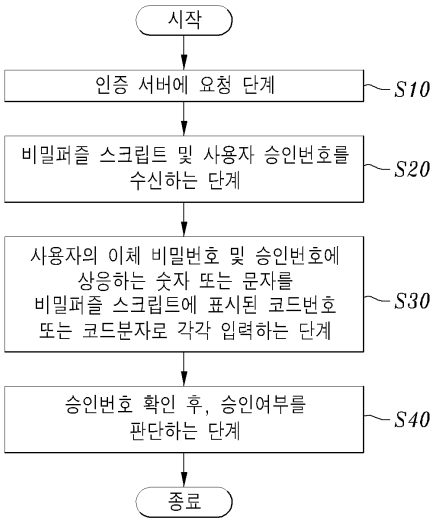
[0177] 본 발명은 상술한 특정의 바람직한 실시 예에 한정되지 아니하며, 청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 당해 발명이 속하는 기술분야에서 통상의 지식을 가진 자라면 누구든지 다양한 변형의 실시가 가능한 것은 물론이고, 그와 같은 변경은 청구범위 기재의 범위 내에 있게 된다.

부호의 설명

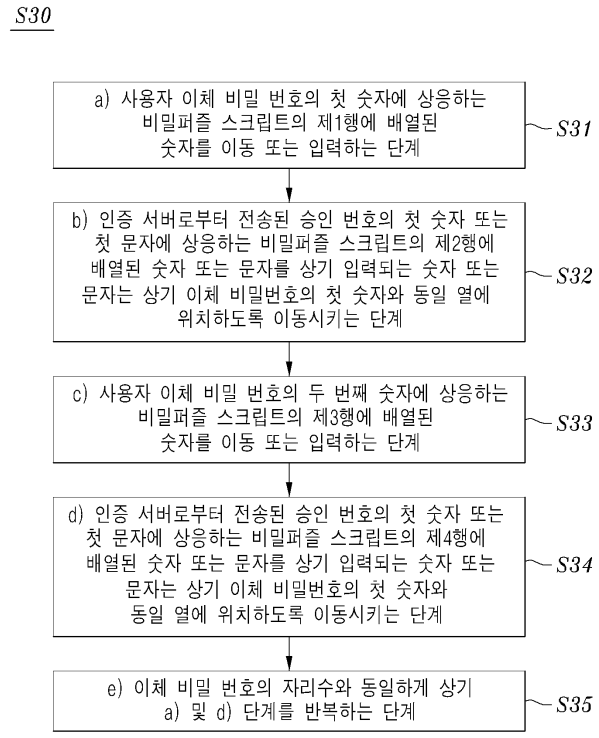
[0178] 10,20,30,40: 비밀퍼즐 스크립트

도면

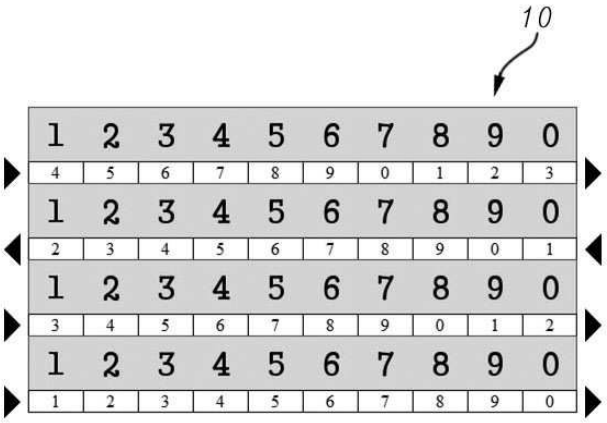
도면1



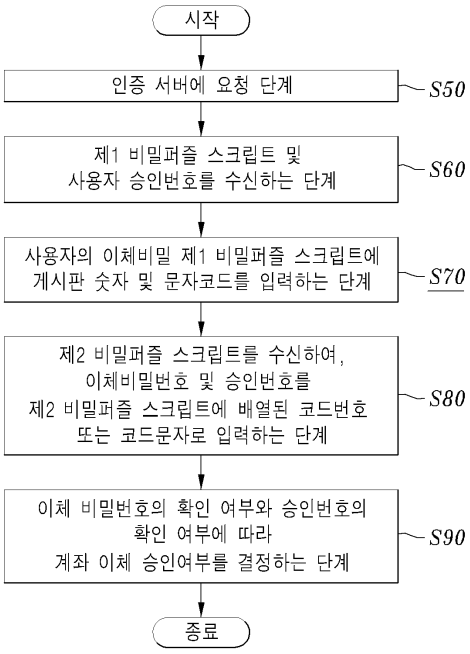
도면2


S30
a) 사용자 이체 비밀 번호의 첫 숫자에 상응하는 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 단계
S31
b) 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계
S32
c) 사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 단계
S33
d) 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 비밀퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계
S34
e) 이체 비밀 번호의 자리수와 동일하게 상기 a) 및 d) 단계를 반복하는 단계
S35

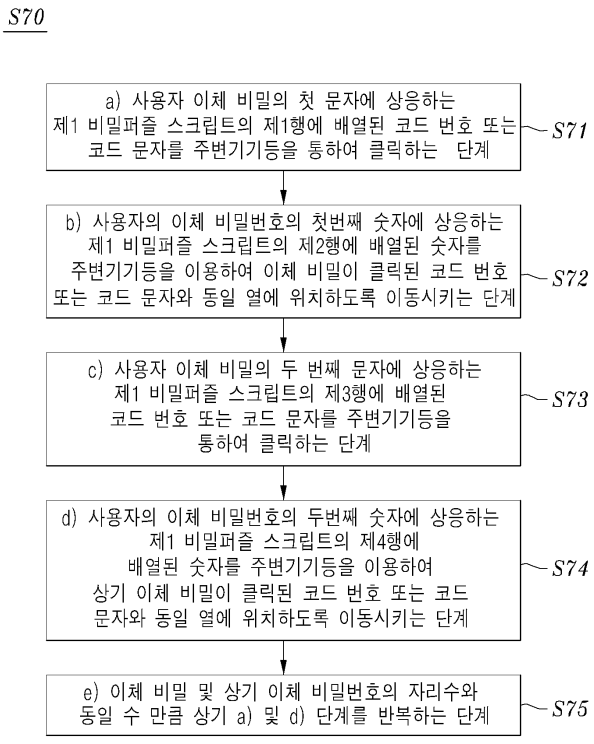
도면3


10
1 2 3 4 5 6 7 8 9 0
4 5 6 7 8 9 0 1 2 3
1 2 3 4 5 6 7 8 9 0
2 3 4 5 6 7 8 9 0 1
1 2 3 4 5 6 7 8 9 0
3 4 5 6 7 8 9 0 1 2
1 2 3 4 5 6 7 8 9 0
1 2 3 4 5 6 7 8 9 0

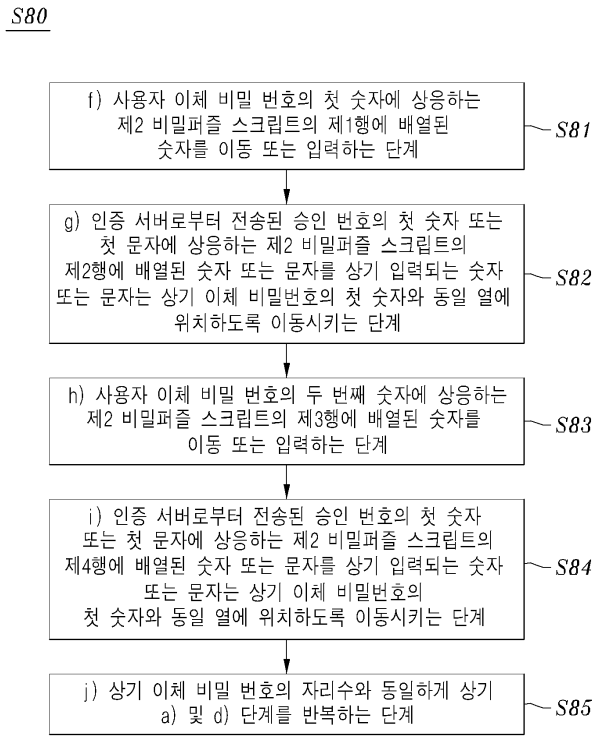
도면4


시작
인증 서버에 요청 단계
S50
제1 비밀퍼즐 스크립트 및
사용자 승인번호를 수신하는 단계
S60
사용자의 이체비밀 제1 비밀퍼즐 스크립트에
게시판 숫자 및 문자코드를 입력하는 단계
S70
제2 비밀퍼즐 스크립트를 수신하여,
이체비밀번호 및 승인번호를
제2 비밀퍼즐 스크립트에 배열된 코드번호
또는 코드문자로 입력하는 단계
S80
이체 비밀번호의 확인 여부와 승인번호의
확인 여부에 따라
계좌 이체 승인여부를 결정하는 단계
S90
종료

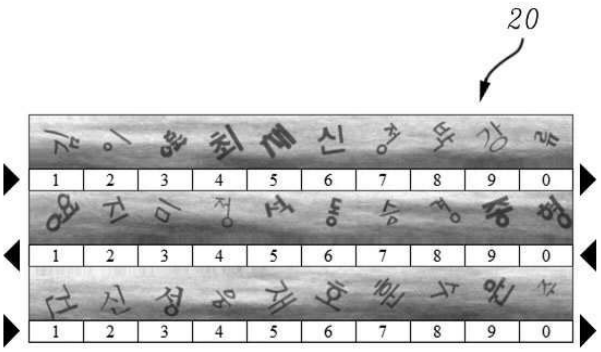
도면5


S70
a) 사용자 이체 비밀의 첫 문자에 상응하는
제1 비밀퍼즐 스크립트의 제1행에 배열된 코드 번호 또는
코드 문자를 주변기기등을 통하여 클릭하는 단계
S71
b) 사용자의 이체 비밀번호의 첫번째 숫자에 상응하는
제1 비밀퍼즐 스크립트의 제2행에 배열된 숫자를
주변기기등을 이용하여 이체 비밀이 클릭된 코드 번호
또는 코드 문자와 동일 열에 위치하도록 이동시키는 단계
S72
c) 사용자 이체 비밀의 두 번째 문자에 상응하는
제1 비밀퍼즐 스크립트의 제3행에 배열된
코드 번호 또는 코드 문자를 주변기기등을
통하여 클릭하는 단계
S73
d) 사용자의 이체 비밀번호의 두번째 숫자에 상응하는
제1 비밀퍼즐 스크립트의 제4행에
배열된 숫자를 주변기기등을 이용하여
상기 이체 비밀이 클릭된 코드 번호 또는 코드
문자와 동일 열에 위치하도록 이동시키는 단계
S74
e) 이체 비밀 및 상기 이체 비밀번호의 자리수와
동일 수 만큼 상기 a) 및 d) 단계를 반복하는 단계
S75

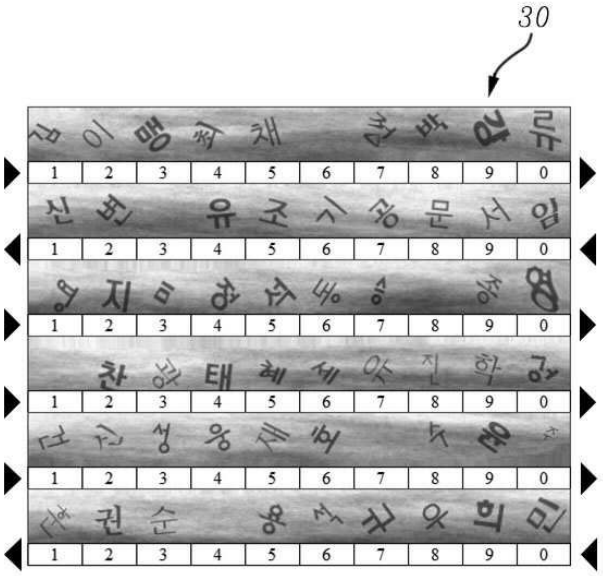
도면6


S80
f) 사용자 이체 비밀 번호의 첫 숫자에 상응하는 제2 비밀퍼즐 스크립트의 제1행에 배열된 숫자를 이동 또는 입력하는 단계
S81
g) 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 제2 비밀퍼즐 스크립트의 제2행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계
S82
h) 사용자 이체 비밀 번호의 두 번째 숫자에 상응하는 제2 비밀퍼즐 스크립트의 제3행에 배열된 숫자를 이동 또는 입력하는 단계
S83
i) 인증 서버로부터 전송된 승인 번호의 첫 숫자 또는 첫 문자에 상응하는 제2 비밀퍼즐 스크립트의 제4행에 배열된 숫자 또는 문자를 상기 입력되는 숫자 또는 문자는 상기 이체 비밀번호의 첫 숫자와 동일 열에 위치하도록 이동시키는 단계
S84
j) 상기 이체 비밀 번호의 자리수와 동일하게 상기 a) 및 d) 단계를 반복하는 단계
S85

도면7a


20
1 2 3 4 5 6 7 8 9 0
1 2 3 4 5 6 7 8 9 0
1 2 3 4 5 6 7 8 9 0

도면7b



도면7c

도면7d