분산 야누스 시스템

조선제*, 이경현**
*부경대학교 전산정보학과 , **부경대학교 전자컴퓨터정보통신공학부
*sjcho@kccips.or.kr, **khrhee@pknu.ac.kr

A Distributed Janus System

Seon-Je Cho*, Kyung-Hyune Rhee**
*Dept. of Computer & Information, PuKyong Nat'l University
**Devision of Electronic, Computer & Telecommunication Engineering, PuKyong Nat'l University

요 약

사용자들이 인터넷상의 웹사이트로부터 정보를 얻고자 할 때, 대부분의 웹사이트는 사용자들의 신분을 확인하고, 권한이 있는 사용자들에게만 정보를 제공한다. 이 때, 사용자들은 접속할려고 하는 웹사이트에 권한이 있음을 확인하는 계정을 만든다. 사용자들이 계정을 만들 때, 웹사이트마다 별개의 계정을 만드는 것이 공격자의 위협으로부터 안전하지만, 현실적으로 사용자의 기억의 한계, 사용자의 편의 추구등으로 인해, 사용자들은 단일 계정으로 여러 웹사이트에 사용할려는 경향이 있다. 따라서, 어느 공격자가 단일 사용자의 특정 웹사이트에 대한 ID와 패스워드를 알게되면, 그 공격자는 다른 웹사이트에 동일한 ID와 패스워드를 사용한 로그인 시도 공격시, 성공할 확률이 매우 높다고 할 수 있다. 본 논문에서는 이러한 사용자 ID와 패스워드에 대한 문제점을 해결하기 위한 방안으로서 사용자는 로그인하려는 웹사이트에 상관없이 항상 동일한 ID와 패스워드를 사용하지만, 실질적으로는 사용자 머신내에 있는 내부 프록시(proxy)에 의해서 웹사이트마다 독립적이고 유일한 가명 ID와 가명 패스워드를 안전하게 생성하는 방안을 제안한다.

1. 서 론

인터넷의 급격한 확장으로 많은 사용자들은 자신들이 필요로 하는 정보를 인터넷상의 여러 웹사이트로 부터 제공받는다. 이 때 웹사이트들은 정보나 서비스를 제공하기 전에, 권한을 가진 사용자들에게만 제공하기 위하여 계정 등록을 요구한다.

사용자가 여러 웹사이트에 계정 등록시 각각 별개의 계정을 등록하는 것은 매우 번거로울 뿐더러 기억하기도 어렵기 때문에 동일한 계정 정보(ID, 패스워드)를 사용하게 된다. 따라서, 어느 공격자가 단일 사용자의 특정 웹사이트에 대한 ID와 패스워드를 알게되면, 그 공격자는 다른 웹사이트에 동일한 ID와 패스워드를 사용한 로그인 시도 공격이 성공할 확률이 매우 높다고 할 수 있다. 본 논문에서는 위와 같은 공격을 간략히 "웹사이트 연결 공격"이라고 언급하겠다.

참고 문헌 [3]에서는 야누스 서버라고 하는 단일 프록시 서버에서 단일 사용자 정보(ID와 패스워드)에 대한 다수의 웹사이트를 위한 가명 ID와 가명 패스워드를 생성함으로써 웹사이트 연결 공격을 방지하기 위한 시도를 하였다. 하지만, 야누스 서버는 시스템 자체가 신뢰되고 공격에 안전해야만 전체 시스템이 정상적으로 운용이 된다. 만약, 어느 공격자가 야누스 서버에 대한 공격을 성공하게 되면, 그 공격자는 야누스 서버를 사용하는 모든 사용자의 유일한 ID와 패스워드를 획득할 수 있을 뿐만 아니라, 각 사용자들의 ID와 패스워드로부터 각 사용자들의 특정 웹사이트를 위한 가명 ID와 가명 패스워드를 자유로이 생성할 수 있게 된다.

따라서, 본 논문에서는 공격의 주요 목표가 되는 야누스 서버에서의 사용자 ID와 패스워드로부터 가명 ID와 가명 패스워드를 생성하기 위한 함수를 임계 암호 기법에 기반하여 좀 더 안전하게 구성하고자 한다. 본 논문의 구성은, 2장에서는 관련 연구에 관하여 기술하고, 3장에서는 제안하는 시스템의 구성 요소에 대해서 설명한 뒤, 4장에서는 이 요소들간의 동작을 살펴본다. 그리고, 5장의 안전성 평가 후 결론을 맺는다.

2. 관련 연구

2.1 야누스 시스템

야누스 시스템[3]에서는 단일 사용자와 여러 웹사이트 사이에 야누스 서버라는 단일 중계 프록시를 두어, 가명 ID와 가명 패스워드를 생성한다. 즉, 사용자가 야누스 서버에 입력하는 단일 사용자 정보로부터 야누스 서버는 다수의 웹사이트에 대한 가명 ID와 가명 패스워드를 생성함으로써 사용자 신원을 보호한다. 하지만, 야누스 서버가 공격자에게 노출 당하면 공격자는 Jauns서버에 입력된 사용자들의 ID와 패스워드를 알게 되고, 이를 이용하여 공격자는 야누스 서버 내의 야누스 함수를 동작시킴으로써 필요한 가명계정들을 자유로이 생성할 수 있는 단점이 있다.

2.2 임계 암호 시스템

(t,n) **임계 암호 시스템** : 공개키 암호 시스템에서의 비밀키를 n개의 참여자에게 비밀 분산시킨다. 따라서, 각 참여자는 위의 비밀값에 대한 비밀 분배값을 소유하게

된다. 만약, 어떠한 비밀 연산이 요구되면, 적어도 t개의 참여자가 협조를 해야만 의미있는 암호적 연산을 수행하는 시스템을 말한다[2][4][5][6][7].

즉, 임계 서명 시스템에서는 어떤 메시지(m)을 서명하기 위해서는 최소 t개의 참여자가 자신의 비밀 분배값으로 계산한 t개의 부분 서명값을 사용하여 분산되어진 원래 비밀값으로 계산한 전자 서명문을 생성한다. 따라서, 공격자가 서명문 위조를 수행하기 위해서는 최소 t개의 참여자들을 공격 성공해야만 한다.

3. 분산 야누스 시스템

본 장에서는 제안하는 시스템의 구성 요소 및 각각의 기능에 대해서 논하고자 한다. 그림 1은 제안 시스템의 구조를 보이고 있다.

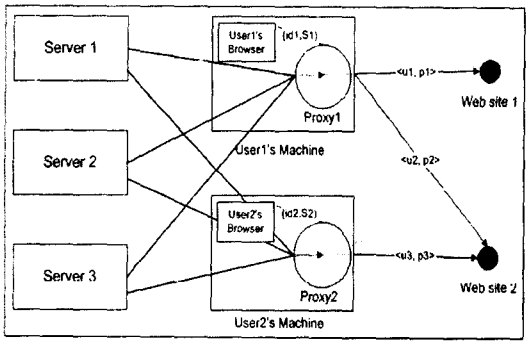

그림 1 분산 야누스 시스템

- **사용자** : 사용자는 자신의 머신 내에 위치한 웹 브라우저를 통하여 웹사이트와 메시지를 주고받는다. 여기서 사용자의 웹 브라우저는 프록시와 자동 연결되고, 프록시에 입력되는 사용자 정보(ID, 패스워드)는 서버들(그림 1에서의 Server1 - Server3)과 프록시로부터 가명계정(가명 ID, 가명 패스워드) 생성에 필요한 변수가 된다.
- **서버** : 서버들은 각각 다른 머신에서 동작되는 하나의 데몬으로 구현되어진다. n개의 서버들에게는 공개키 암호 알고리즘을 위한 비밀키(secret key)가 [1]의 기법을 사용하여 분산(sharing)되어, 각 서버들은 자신의 비밀 분배값(private shared key)을 소유하게 된다. 프록시로부터 부분 서명값 요청이 있을 때 각 서버들은 자신의 비밀 분배값을 이용한 부분 서명값을 전달한다.
- **프록시(Proxy)** : 프록시는 사용자 머신 내에 위치하며, 서버들과의 안전한 채널을 형성한다. 프록시는 같은 머신 내의 사용자의 정보를 입력받는다. 그 후, 특정 사이트로부터 가명계정의 요청이 있을 경우 서버들에게 해당 사용자 정보(ID, 패스워드)와 특정 웹사이트에 대한 정보를 전달하고, 서버들로부터 수신한 부분 서명값으로 서명값을 유도하여 가명계정을 생성한다.

4. 가명 생성 시스템의 동작 방식

본 장에서는 3장에서 기술된 구성요소들 간의 상호 동작 방식에 대해 살펴본다.

4.1. 동작 시나리오

각 사용자의 브라우저는 같은 머신내의 프록시와 연결되어 있고, 웹 브라우저에서의 모든 메시지는 프록시를 거쳐가도록 설정된다. 그리고, 서버들과 프록시와의 모든 통신 메시지는 안전하게 전달되어 진다고 가정한다.

그림 1에서는 두 사용자의 경우를 예를 들어 전체 시스템을 구성하였다. $User1$은 $w1$이라는 웹사이트에 접속을 시도한다고 가정한다. 이 때 $Proxy1$은 자동 실행되어 $User1$의 정보 $(id1, S1)$를 입력받은 후, $w1$으로 이동되게 한다. 여기서 $id1$은 $Proxy1$에 대한 $User1$의 ID를 나타내고, $S1$은 해당 ID에 대한 일종의 패스워드를 나타낸다. 이렇게 $User1$의 정보 입력은 $Proxy1$을 가동시킨다. 그리고, 가동되는 동안 $Proxy1$은 사용자의 요청에 따라 계속해서 가명계정($<u_i, p_i>$)을 생성하게 된다.

$Proxy1$의 동작은 $User1$이 $w1$의 등록폼이나 로그인폼에 입력하는 ID와 패스워드 값에 따라 다음과 같이 나뉜다.

- 실제로 사용하고자 하는 ID와 패스워드를 입력한다면, $Proxy1$은 동작하지 않고 $User1$은 자신이 입력한 계정으로 $w1$을 이용하게 된다.
- ID 입력 부분에는 "/U"를, 패스워드 입력 부분에는 "/P"를 입력한다면, $Proxy1$은 그 문자에 반응하여 서버들로부터 $User1$의 입력 정보(ID와 패스워드)와 해당 웹사이트 정보에 대한 부분 서명값들을 요청하게 되고, (t,n) 임계 암호 방식에 따라서 t개의 유효한 부분 서명값으로 부터 서명값을 계산한다. 이 때 이 서명값은 가명계정의 생성 정보가 된다.

그리고, $User1$의 $w2$에 대해서는 $w1$과는 별개의 가명계정(그림1 의 $<u2, p2>$)을 생성하게 된다. 세션이 중단된 뒤 새로 시작되더라도 특정 사용자의 특정 사이트에서의 가명계정은 항상 동일하게 생성된다.

4.2. 가명계정 생성 프로토콜

이 절에서는 서버들과 프록시 사이에 가명계정이 어떻게 생성되는지에 대해 기술한다. 본 논문에서 사용되는 표기법은 다음과 같다.

- Pn: ID가 n인 Proxy
- λ_X : 사용자 X의 랜덤 비트열. 사용자 X는 임의의 랜덤 비트열 λ_X를 생성 시스템에 안전하게 보관한다. 예를 들어, $User1$은 자신의 $S1$을 암호화 키로 하여 λ_{User1}값을 암호화하여 시스템에 저장하고 있다.
- Sn: ID가 n인 Server
- $SIG_X(\cdot)$: 통신개체 X의 개인키를 사용한 전자서명
- I_X: 통신개체 X의 인덱스(index)로 기본값은 0이며, 웹사이트로부터 계정 변경을 요청 받을 경우에 사용되는 변수

프로토콜 가정 사항 : $P1$과 서버들(Sn)간은 Secure

Socket Layer(SSL)을 통하여 키 교환과 상호인증이 형성되어 있고, $User1$은 $P1$에 ID와 패스워드를 입력하여 $P1$을 활성화 시킨 상태이다. 따라서, $User1$의 λ_{User1} 값 또한 복호되어서 $P1$에 로드되어져 있는 상태이다. 또한, 제안 기법에서 추상적으로 사용되는 임계 전자 서명기법에서 기존에 확률적 트랩도어 일방향 함수 성질을 기반하지 않고 있다. 즉, RSA의 경우 PKCS#1이나 OAEP 패딩과 같은 랜덤 패딩이 없이 단순한 정수론적 서명 연산을 고려한다.

[Step1] 만약 $User1$이 가명 ID와 가명 패스워드를 사용하고자 할 때, $User1$은 웹브라우저를 통하여 $w1$의 ID입력부분에는 "/U", 패스워드 입력 부분에는 "/P"의 입력으로, $P1$에게 가명계정 생성을 요청한다.

[Step2] $P1$은 n개 서버들에게

$$INFO = (id1 \| S1 \| w1 \| I_{w1} \| \lambda_{User1})$$

값을 전달하여, 각 서버들의 부분 서명값을 요청한다.

[Step3] 각 서버들 $(S_i,\ 1 \le i \le n)$ 은 자신의 비밀 분배값을 이용하여 아래와 같은 부분 서명값을 계산하여, $P1$에게 전달한다.

$$SIG_{Si}\ (INFO)$$

[Step4] $P1$은 n개의 서버들로부터 받은 t개의 유효한 부분 서명값들로부터, 유효한 전자 서명값을 유도한다. 여기서, 유효한 전자 서명값, $SIG(INFO)$을 유도하기 위한 함수(Combining function)은 적용되는 임계 암호 시스템[1][4][5]에 의존적인 알고리즘이다.

[Step5] $P1$은 Step4에서 생성된 전자 서명값에 일방향 해쉬함수에 기반한 PRF(Pseduo Random Function)를 적용시켜서 필요한 가명 ID와 가명 패스워드를 생성한다.

$$PRF(SIG(INFO)) = (u1 \| p1)$$

[Step6] 결국, $P1$은 생성한 가명 ID와 가명 패스워드를 $w1$에게 전송한다.

만약, 어떤 웹사이트로부터 계정 변경 요구가 있을 경우에 아래와 같은 형태로 변경 가능하다.

[계정 갱신 기법]

$User1$이 $w1$에 이미 가명계정을 통하여 등록되어있다고 가정하자. 만약, 사용자가 $w1$으로부터 계정 변경 요청을 받게 되면, $P1$은 $User1$에 대한 I_{w1}의 값을 a만큼 증가시킨 후, 변경된 $User1$의 정보($INFO$)로 새로운 부분 서명값을 서버들에게 재요청하여, 다른 사이트의 가명계정의 변경없이 해당 사이트의 새로운 가명계정을 생성받도록 한다.

5. 안전성 및 평가

제안 방안은 다음과 같은 암호학적 강도를 지닌다.

- 강건한 가명생성 함수: 기존의 야누스 서버를 통한 가명 생성 시스템에 임계 암호 기법을 적용하여 새로이 설계한 시스템으로, 시스템 자체가 신뢰되어야만 정상적인 운영이 가능한 기존 방식[3]과는 달리 $t-1$ 개까지의 서버들의 노출(compromise)을 허용함으로 좀 더 공격에 유연성이 있도록 설계하였다.
- 웹사이트 연결 공격 방지 : 가명 생성 정보가 프록시만으로 생성되어지는 것이 아니라 (t, n) 임계 암호 시스템에 근거한 서버들의 도움이 있어야만 하므로, 단지 사용자 머신의 공격을 통해 사용자 정보와 특정 사이트에 대한 서명값을 획득하더라도, 다른 사이트에 대한 가명계정 생성에 필요한 정보들은 알 수 없으므로 웹사이트 연결 공격이 불가능해진다.
- 유일한 가명 계정의 보장 : 동일한 실제 ID와 패스워드를 가진 사용자들은 λ 값을 통해서 동일한 웹사이트에 대해서도 서로 다른 가명 계정을 소유하게 된다.

6. 결 론

본 논문에서는 사용자 신원을 보호하기 위한 새로운 가명계정 생성 시스템을 제안하였다. 제안 방안은 기존 야누스시스템과 임계 암호 시스템과의 결합으로 공격에 대한 유연성과 안전성을 지니는 새로운 가명계정 생성 방식이다. 향후 구현을 위해 구체적이고 세부적인 네트워크 및 시스템의 정상적인 운영 변수를 고려함으로써 다양한 공격에 대처 가능한 시스템의 개발이 적 할 것으로 판단된다.

[참고문헌]

[1] A. Shamir. "How to share a secret", Communications of the ACM, 22:612-613, 1979.

[2] D. Boneh, M. Franklin, "Building Intrusion Tolerant Applications", in Proceedings Crypto'97 pp. 425-439

[3] Eran Gabber, Phillip B. Gibbons, Yossi Matias, Alain Mayer, "How to Make Personalized Web Browsing Simple, Secure, and Anonymous", In Proceedings of Financial Cryptography '97, 1997.

[4] M. Malkin, T. Wu, D. Boneh, "Experimenting with Shared Generation of RSA keys", in Proceedings of the Internet Society's 1999 Symposium on Network and Distributed System Security(SNDSS), pp. 43-59

[5] T.Rabin, "A simplified approach to threshold and proactive RSA", Proceedings of Crypto'98, pp. 89-104

[6] Victor Shoup, "Practical Threshold Signatures", in Proceedings Eurocrypt 2000

[7] Yvo Desmedt, "Some Recent Research Aspects of Threshold Cryptography", In information Security, First International Workshop ISW '97, volume 1196 of Lecture Notes in Computer Science, pp. 158-173, 1997.