네트워크 통신에서 패킷의 암호화 알고리즘의 키 해킹시 데이터 보안을 위한 조립알고리즘 제안

장승철*, 전문석*
*숭실대학교 대학원 컴퓨터학과
E-mail : lagris@naver.com, mjun@ssu.ac.kr

Suggest to Organization Algorithm for Data Security when Hacking to Encryption Algorithm's key in Network Data Communication

Jang SeungChul*, Jun MoonSuk*
*Dept of Computer Science Soongsil Universit
E-mail : lagris@naver.com, mjun@ssu.ac.kry

요 약

오늘날 네트워크 통신은 다양한 해킹방법들로부터 데이터를 보호하기 위하여 다양한 암호화 알고리즘을 사용하여 통신하고 있다. 이렇게 사용되는 암호화 알고리즘에서 키가 누출되어 해킹되었을 경우 데이터에 대한 정보누출 및 수정, 삭제 등의 데이터 가공을 당하게 된다. 본 논문에서는 이러한 문제를 보완하기 위하여 본래의 데이터를 섞고, 재조립하는 조립알고리즘을 제안한다.

1. 서론

최근 네트워크 사용자의 폭발적인 증가와 사용자들이 사용하는 인터넷 서비스들의 증가로 사용자의 개인정보 등의 보안문제가 중요시 되게 되었다. 특히 인터넷 게임과 같이 많은 사용자가 데이터를 주고받는 인터넷 서비스에서는 극소수의 사용자가 해킹 등의 불공정 서비스를 이용하더라도 해킹 도구의 온라인 전파나 보안 결함의 대중적 여론 악화 등으로 해당 온라인 서비스는 치명적 손실을 입게 된다.

이러한 문제는 사용자 정보, 사용자 인증 등과 같은 정보유출 문제 등을 발생시키고, 해당 온라인 서비스의 여론 악화 및 신뢰도의 하락 등을 가져오게 된다. 이러한 문제에 대한 기본적인 해결책은 다른 정보 보호 분야에서와 마찬가지로 데이터를 암호화하는 것이다.[1]

본 논문에서는 인터넷 서비스에서의 암호화 통신에서 사용되는 키가 해킹으로 인해 누출 되었을 경우 해당 인터넷 서비스의 암호화된 데이터를 복호화하여 데이터의 누출 및 수정 삭제 등의 악의적으로 사용될 수 있으므로 송신측에서 암호화하기 이전의 원본 데이터를 섞고 암호화하여 전송하고 수신측에서 복호화하고 해당 패킷을 재조립하는 조립 알고리즘을 제안한다.

2. 관련연구

2.1. 암호의 종류

일반적으로 암호화를 하기 위해서는 사용자의 암호화 키가 필요하고 복호화를 위해서는 복호화 방법과 복호화 키가 필요하다.

[Table 1] 암호화시스템

암호방식 / 항목	대칭키 암호방식	공개키 암호방식
암호화 키	비밀	공개
복호화 키	비밀	비밀
암호알고리즘	비밀/공개	공개
비밀키 전송	필요	불필요
암호화 속도	고속	저속

대칭키 알고리즘은 비밀키 알고리즘이라고도 하며, 송신자와 수신자가 같은 비밀키를 서로 공유하고 있는 것을 전제조건하에 이루어지는 방식이다. 비밀키

암호는 송신자와 수신자를 제외한 삼자에게 알려져선 안되고 구성원이 많을수록 비밀키의 공유는 어려워 진다.

비대칭키 알고리즘은 공개키 암호 알고리즘이라고도 하며, 송신자와 자신만의 비밀키를 이용하여 암호화시키면 수신자는 송신자의 공개키를 이용하여 복호화하는 방식이다. 이때, 공개된 암호화키로부터 복호화키를 알아낼 수 없어야한다.[2]

2.2. 블록 암호화와 스트림 암호화

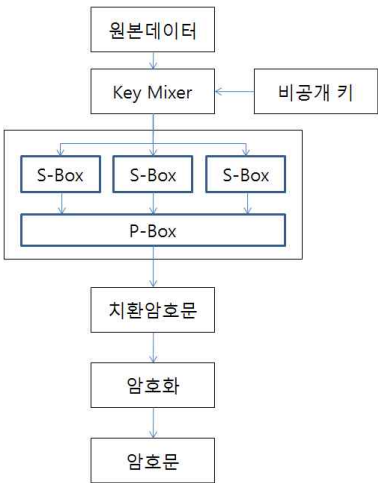
블록 암호화 방법은 비트열로 입력되는 데이터를 일정한 길이의 비트열로 잘라 그 단위로 암호화 하는 방법으로 한 블록이 변조 되어도 다른 블록까지 영향을 미치지 않는다.

스트림 암호화 방법은 비트열로 입력되는 데이터를 비트의 단위로 암호화 하여 전송하는 방법으로, 어떤 한 비트가 변조되면 전체적인 암호문이 바뀌게 되는 암호화 방법이다. 스트림 암호화는 Vigenere 표를 사용하는 Vigenere Cipher의 수정된 암호화 방법으로, 원본 데이터가 도착하면 출력 레지스터의 8 Bit 논리연산, 즉 Ex-OR가 되어 암호문이 생성된다.[3]

3. 대치와 치환을 이용한 조립 알고리즘 설계

본 논문에서 제안하고자 하는 조립 알고리즘은 치환(Permutation) 암호를 응용한 기법으로 키의 누출에 대한 데이터의 보호와 이중적인 암호화에 대한 부하를 줄이고자 하였다.

3.1. 암호화 과정



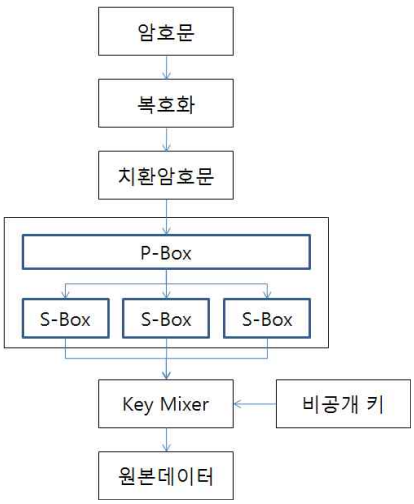
[그림 1] Client Flow Chart

암호화 과정을 설명하는 것은 다음과 같다.

1) 전송하고자 하는 원본 데이터를 일정한 길이로 자른다. 해당 블록을 공개되지 않는 키와 함께 Key Mixer를 통하여 XOR연산을 한다.
2) Key Mixer를 통하여 나온 XOR된 값을 S-Box와 P-Box를 통하여 암호문을 추출한다.
3) 추출된 암호문을 통하여 대칭키 또는 비대칭키 암호화를 행한다.
4) 암호화를 통해 나온 암호문을 전송한다.

위의 암호화과정에서 문자를 입력받은 값은 이진화하여 비공개키와 함께 Key Mixer를 통해 XOR 연산을 행하고, S-Box와 P-Box를 통해 대치와 치환을 행하여 기존의 암호화와 더해 2중의 암호문을 생성하는 것이다. 위의 과정에서 비밀키는 외부에 공개되지 않고 송신자와 수신자에만 공개되는 것을 전재로 한다.

3.2. 복호화 과정



[그림 2] Server Flow Chart

복호화 과정을 설명하는 것은 다음과 같다.

1) 송신측으로부터 전송되어진 암호문을 기존에 사용하던 방법으로 복호화 한다.
2) 복호화된 치환암호문을 P-Box와 S-Box를 이용하여 대치와 치환을 시킨다.
3) 송신자와 수신자에게만 공개되어있는 비공개 키와 대치와 치환을 거친 값을 Key Mixer를 통하여 XOR 연산을 실행한 후 원본 데이터를 추출한다.

위의 복호화 과정에서도 비공개 키는 송신자와 수신자만 공개되고 3자에게는 공개되지 않는다는 전제가 있다.

4. 결론

본 논문에서는 다수가 접속하는 대규모의 인터넷 서비스에서 사용하는 암호화 방법 중 해킹을 통하여 키와 알고리즘이 공개되었을 경우를 위하여 데이터의 보안을 강화하기 위해 대치와 치환을 이용하여 원본 데이터를 암호화를 하였다.

위의 암호화 방법은 데이터에 대한 암호화를 강화하고 서비스에 대한 부하를 크게 늘리지 않는 방법으로 유용할 것이다. 또한 알고리즘이 간단하기에 단순하면서도 다양한 방법으로 활용할 수 있을 것이다.

참고문헌

[1] 노지명, 양정민, “보안 요구 수준에 근거한 효율적인 패킷 암호화 방법”, 한국정보보호학회, 정보보호학회논문지 Vol.14 No.5, pp. 153-164, 10월, 2004.

[2] 권성택, “데이터 전송시의 데이터 보안을 위한 암호화 방법에 관한 연구”, 우암논총학술지, 제16권, 1997년

[3] 장신도, 최영규 “대칭형 암호기술을 응용한 패킷 암호화에 관한 연구”, 한국정보기술학회, 한국정보기술학회 하계종합학술발표논문집 2005년도 하계종합학술발표논문집, pp. 152-156, 7월, 2005년