해시 알고리즘의 안전성 동향

홍남수*, 강정호*, 전문석*
*숭실대학교 컴퓨터학과
e-mail:sucream@ssu.ac.kr

Safety Trend of Hash Algorithm

Namsu Hong*, Jungho Kang*, Moon-Soeg Jun*
*Dept. of Computer Science & Engineering, Soongsil University

요 약

해시 함수는 데이터의 위변조를 확인하기 위해 사용하는 일방향 함수로, 현재 많은 기술 및 논문에서 해시 함수를 사용하고 있다. 대표적인 해시 함수에는 MD와 SHA가 있으며 다양한 버전을 가지고 있다. 본 논문에서는 해시 알고리즘들의 안전성과 관련된 동향과 취약점을 파악하고 향후 방향성을 알아보고자 한다.

1. 서론

해시 알고리즘은 하나의 문자열을 고정된 길이의 값이나 키로 변환하는 것이다. 해시 알고리즘은 암호화와 비슷하지만 다른 특징을 가지고 있다. 일반적인 암호화는 정보를 숨기는 것이 주된 목적이라면, 해시 알고리즘은 데이터의 무결성을 지키는 것이 목적이다. 해시 알고리즘은 평문의 길이와 상관없이 결과 값의 길이가 모두 같고 평문이 조금만 달라지더라도 결과를 추측하는 것이 불가능해야 한다. 즉, 서로 다른 두 해시 값이 존재한다면 각 해시 값에 해당하는 원본 데이터도 달라야 한다. 현대 사회에서 해시 알고리즘은 대칭, 비대칭 암호화 기법과 함께 사용되며 전자서명, 전자봉투, 전자화폐 등 다양한 곳에서 사용되고 있다. 만약 해시 알고리즘의 취약점으로 인해 상이한 데이터들이 동일한 해시 값을 가지게 된다면 심각한 문제가 발생할 수도 있다. 본 논문에서는 다양한 해시 알고리즘 중에서 암호학적 해시 알고리즘으로 사용되는 대표적인 알고리즘인 Message-Digest(MD)와 Secure Hash Algorithm(SHA)에 대한 동향 및 취약점을 살펴보고 향후 해시 알고리즘의 방향성에 대해 논의하고자 한다.

2. 해시 알고리즘의 보안 요구사항

암호학적 해시 함수는 Preimage resistance(역상 저항성), Second preimage resistance(두 번째 역상 저항성), Collision resistance(충돌 저항성) 세가지를 만족해야 한다.[1] Preimage resistance는 주어진 해시 함수 h와 y = h(M)에 대해서 악의적인 사용자가 y = h(M')를 만족하는 메시지 M'을 찾아내는 것이 매우 어려워야 한다는 성질이다. Second preimage resistance는 악의적인 사용자가 원래의 메시지인 M과 해시한 값인 h(M)을 알 때, h(M) = h(M')을 만족하는 다른 메시지 M'을 생성해내는 것이 매우 어려워야 한다는 성질이다. Collision resistance는 악의적인 사용자가 아무런 정보 없이 h(x) = h(y)를 만족하는 두 개의 메시지 x, y를 찾아내는 것이 매우 어려워야 한다는 성질이다.

3. Message-Digest(MD)

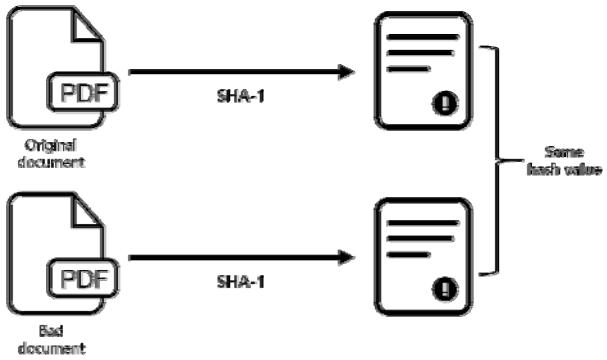
MD는 MD2, MD4, MD5, MD6 총 4가지 버전이 존재한다. MD의 최종 버전은 MD5로, Rivest Shamir Adleman(RSA) 알고리즘을 개발한 Rivest 교수가 1991년에 개발하였다. MD5는 입력된 메시지를 512비트로 된 블록들로 나누고 128비트의 다이제스트를 출력한다. MD5는 일반적인 해시 알고리즘들과 마찬가지로 단방향 암호화를 수행하며, 출력값을 통해 원래 값을 복원할 수 없다. MD5는 보통 패스워드 암호화, FTP를 통해 송수신되는 데이터의 무결성을 검증하기 위해 사용되었다. 하지만 MD5는 2004년에 높은 유사성을 보이지만 다른 두 파일의 해시값의 충돌이 발견되었고, 2006년에는 노트북(Intel Pentium 1.6GHz) 한 대 만으로 1분 만에 충돌을 찾아냈다.[2] 따라서 현재 MD는 권장되어지는 알고리즘이 아니기 때문에 다른 안전한 알고리즘을 사용할 것을 권장하고 있다. MD는 네트워크로 전송되는 큰 파일의 무결성을 확인하거나 보안용도로 사용시 반드시 salt값을 붙여서 사용해야 안전하다. MD6의 경우, National Institute of Standard and Technology(NIST)에서 SHA-2를 대체하기 위한 해시 알고리즘 공모에 제출되었지만 속도면에 있어서 불안정하다는 판정을 받고 탈락을 하게 되었다.[3] 현재 MD6는 부분적으로 사용되어지고 있다.

<표 1> MD와 SHA의 종류 및 특징

알고리즘	출력 비트 수	블록 크기	라운드 수
MD2	128	128	864
MD4	128	512	48
MD5	128	512	64
SHA-0	160	512	80
SHA-1	160	512	80
SHA-256/224	256/224	512	64
SHA-512/384	512/384	1024	80

4. Secure Hash Algorithm(SHA)

SHA는 1993년부터 National Security Agency(NSA)에서 개발하고 NIST에서 표준으로 지정한 해시 암호 알고리즘이다. 1993년에 SHA-0를 시작으로 현재 2001년에 나온 SHA-2를 사용 중이다. 완전한 알고리즘은 존재하지 않기 때문에 시간이 지남에 따라 SHA-1의 다양한 취약점이 발견되고 해독 가능성이 제시되었다.



(그림 1) SHA-1의 같은 해시 값을 가지는 충돌

결국 2008년 SHA-1의 해시 충돌이 발생하였으며, 2015년 말기부터는 HTTPS 통신을 위한 디지털 인증서의 해시 값으로 사용하던 SHA-1을 주요 브라우저들이 지원을 중단할 것이라고 선언하였다. 또한 2017년 2월에 구글에서 SHA-1의 충돌 현상을 입증했다. 이 연구에서 중요한 점은 우연한 충돌의 발생이 아니라, 서로 다른 PDF문서를 고의적으로 같은 SHA-1 값을 가지게 했다는 것이다. 구글은 약 90일 후에 충돌 코드를 공개하겠다고 밝혔다. 이는 현재 SHA-1을 사용하고 있는 기업 및 웹 사이트에서 구글이 코드를 공개하기 전까지 안전한 해시 알고리즘으로 교체하지 않을 경우 취약한 상태에 노출된다는 것을 뜻하고 있다. 또한 SHA-1과 SHA-2는 Message Digest의 길이는 다르지만 전반적인 구조가 동일하기 때문에 이번 SHA-1의 충돌 공격이 SHA-2에서 안전하다는 보장이 없다. NIST는 이를 위해 2012년에 SHA-3로 선정할 알고리즘 공모에서 Keccak이라는 새로운 알고리즘을 선정하였다.[4] Keccak은 기존 SHA-1과 SHA-2 두 알고리즘과 설계 자체가 다르기 때문에 SHA-2에서는 효과적일 수 있는 공격이라도 Keccak에서는 거의 효과가 없을 것이라고 밝혔다.

5. 결론

해시 알고리즘은 빠른 검색을 위한 색인으로의 사용, 데이터의 무결성을 위한 암호학적 사용 등 다양한 분야에서 사용되어 오고 있다. 하지만 시대가 지남에 따라 컴퓨터 및 디바이스의 성능이 좋아지고, 다양한 연구들의 결과로 해시 알고리즘들의 다양한 취약점이 발견되었다. MD5의 취약점으로 인해 SHA-1을 사용하게 되었고, 최근 SHA-1의 취약점으로 인해 SHA-1 뿐만 아니라 SHA-1과 비슷한 SHA-2 역시 잠재적인 취약점이 존재할 것이라는 불안감이 고조되고 있다. 본 논문에서는 MD와 SHA에 대한 동향 및 취약점, 이후 방향성을 알아보았다. 해시 알고리즘은 다양한 분야에서 앞으로도 계속 사용될 것이기 때문에 더욱 많은 연구와 질 높은 연구가 필요할 것으로 보인다.

참고문헌

[1] Hoffman, P., and B. Schneier. "RFC 4270: Attacks on cryptographic hashes in internet protocols." Network Working Group (2005).

[2] Sotirov, Alexander, et al. "MD5 considered harmful today, creating a rogue CA certificate." 25th Annual Chaos Communication Congress. No. EPFL-CONF-164547. 2008.

[3] Rivest, Ronald L., et al. "The MD6 hash function - a proposal to NIST for SHA-3." Submission to NIST 2 (2008): 3.

[4] Boutin, Chad. "NIST selects winner of Secure Hash Algorithm (SHA-3) Competition." Press release., October 2 (2012).