본래 Rijndael이라는 이름으로도 알려진 AES는 2001년 미국 국립표준기술연구소(NIST)가 제정한 전자 데이터의 암호화를 위한 규격이다.

AES는 NIST에 AES선발과정 중 방법을 제출한 Joan Daemen과 Vincent Rijmen이라는 두 벨기에 암호학자가 개발한 Rijndael block cipher의 변형이다. Rijndael은 키와 블록 사이즈가 다른 암호의 한 계열이다. AES에 경우, NIST는 Rijndael 계열의 세 구성원을 선택했는데, 각각 블록 사이즈가 128비트지만 키 길이는 각각 128, 192, 256비트이다.

AES는 미국 정부에 의해 채택되었다. 1977년에 발표된 데이터 암호화 표준(DES)을 대체한다. AES에 기술된 알고리즘은 대칭 키 알고리즘으로, 데이터를 암호화하고 복호화하는 데 동일한 키가 사용된다는 것을 의미한다.

미국에서, AES는 2001년 11월 26일에 NIST에 의해 U.S. FIPS PUB 197 (FIPS 197)으로 발표되었다. Rijndael 암호가 가장 적합한 것으로 선택되기 전, 5년간 15개의 암호 후보를 평가하는 과정을 거쳤다.

AES는 ISO/IEC 18033-3 표준에 포함되어 있다. AES는 미국 상무부 장관의 승인을 거쳐 2002년 5월 26일 미국 연방정부 표준으로 발효되었다. AES는 다양한 암호화 패키지로 제공되며, NSA가 미국 국가안보국(NSA)이 극비 정보를 암호화 하기 위해 승인한 최초의(그리고 유일한) 공개적으로 이용 가능한 암호이다.

1. 분류 기준

The Advanced Encryption Standard (AES)는

FIPS PUB 197: Advanced Encryption Standard (AES)

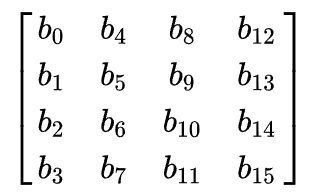
ISO/IEC 18033-3: Block ciphers

로 불린다.

2. 암호 작동 설명

AES는 치환-순열 네트워크로 알려진 설계 원리에 기초하고 있으며 소프트웨어와 하드웨어 모두에서 효율적이다. 이전 DES와 달리 AES는 파이스텔 구조를 사용하지 않는다. AES는 128비트의 고정 블록 크기와 128, 192, 256비트의 키 크기를 가진 Rijndael의 변형이다. 대조적으로, Rijndael 자체는 32비트의 배수가 될 수 있는 블록과 키 크기로 지정되며, 최소 128비트와 최대 256비트가 있다. 대부분의 AES 계산은 특정 유한 필드에서 수행된다.

AES는 다음과 같은 16바이트의 4x4 열 우선 배열로 동작한다:



AES 암호에 사용되는 키 크기는 평문이라고 하는 입력값을 암호문이라고 하는 출력값으로 변환하는 변환 라운드 수를 지정한다. 라운드 수는 다음과 같다:

-128비트 키의 10라운드

-192비트 키의 12라운드

-256비트 키의 14라운드

각 라운드는 암호화 키에 따라 달라지는 단계를 포함하여 여러 처리 단계로 구성된다. 동일한 암호화 키를 사용하여 암호문을 원래 평문으로 다시 변환하기 위해 일련의 역 라운드가 적용된다.

2-1. 알고리즘에 대한 개요

(1)KeyExpansion: 라운드키는 AES 키 스케줄을 이용한 암호화 키에서 만들어졌다. AES는 각 라운드+1 개수만큼의 서로 다른 128비트의 라운드 키블록을 필요로한다.

(2)첫 라운드키 추가:

(2-1) AddRoundKey: 행렬의 각 라운드와 라운드키를 XOR하여 결합한다.

(3)1~9라운드(or 11 or 13)

(3-1)SubBytes: S-Box에 따라 각 바이트가 다른 바이트로 대체되는 비선형 치환 단계

(3-2)ShiftRows: 행렬의 마지막 세 행이 일정 수의 단계를 주기적으로 이동하는 변환 단계

(3-3)MixColumns: 각 열의 4바이트를 결합하여 행렬의 열을 연산하는 선형 혼합 연산

(3-4)Addroundkey

(4)마지막 라운드

(4-1)SubBytes

(4-2)ShiftRows

(4-3)AddRoundKey

2-2. SubBytes 단계

SubBytes 단계에서 행렬의 각 바이트는 8비트 S-Box를 사용하여 SubByte S(a\_i,j)로 교체된다. 0 라운드 이전의 행렬 배열은 단순 평문(입력값)이다. 사용되는 S-Box는 적절한 비선형 특성을 갖는 것으로 알려진 GF(2^8)곱셈의 역원에서 파생된다. 단순 대수적 특성 기반 공격을 피하기 위해, S-Box는 역함수와 가역 아핀 변환을 결합하여 구성된다. 또한 S-Box는 고정점과 반대고정점을 피하기 위해 선택된다. (즉 그래서 완전순열이다.)

복호화를 수행할 때, InvSubBytes 단계(SubBytes의 역순)이 사용되며, 아핀 변환의 역수를 취한 다음 곱셈의 역원을 찾아야 한다.

2-3. ShiftRows 단계

ShiftRows 단계는 행렬의 행에서 작동하며, 각 행의 바이트를 특정 오프셋만큼 주기적으로 이동한다. AES의 경우 첫 번째 행은 변경되지 않은 상태로 유지된다. 두 번째 행의 각 바이트는 왼쪽으로 하나씩 이동된다. 마찬가지로, 세 번째와 네 번째 행은 각각 2와 3 만큼 이동된다. 이러한 방식으로 ShiftRows 단계 출력문의 각 열은 입력문 각 열의 바이트로 구성된다.

이 단계는 각각의 열이 서로 따로 암호화되지 않도록 하는 것이며, 단계가 없어질 경우 4개의 블록 암호화를 실행한 것과 마찬가지가 된다.

2-4. MixColumns 단계

MixColumns 단계에서 행렬의 각 열 4바이트는 가역 선형 변환을 사용하여 결합된다. MixColumns 함수는 4바이트를 입력값으로 사용하고 4바이트를 출력하며, 각 입력 바이트는 출력값에 영향을 미친다. ShiftRows와 함께 MixColumns는 암호에서 확산을 제공한다.  
이 작업을 수행하는 동안 각 열은 다음 고정 행렬을 사용하여 변환된다(행렬에 열을 왼쪽으로 곱하면 행렬의 새 열 값이 표시된다.).

텍스트, 시계이(가) 표시된 사진

자동 생성된 설명

행렬 곱은 곱셈과 덧셈의 성분으로 구성되어 있다. 성분들은 x^7차수의 다항식 계수로 처리되는 바이트들이다. 덧셈은 단순히 XOR이다. 곱셈은 기약 다항식 x^8+x^4+x^3+x+1을 이용한 모듈라 곱셈이다. 비트단위로 처리된 경우, 시프트 후, 시프트된 값이 FF(16진법)보다 크면 1B로 조건부 XOR을 수행해야 한다. (생성된 다항식의 차감을 통해 오버플로를 보정해야 한다.) 이것은 GF(2^8)에서 일반적으로 사용되는 곱셈의 특수한 경우이다.

좀 더 일반적인 의미에서, 각 열은 GF(2^8)의 다항식으로 처리되며, a(x)=3x^3+x^2+x+2를 곱한 값을 x^4+1로 모듈러 곱셈을 한다. 계수는 GF(2^8)의 비트 다항식의 2진 표현에 해당되는 16진수로 표시된다. 또한 Mixcolumns 단계는 유한 필드 GF(2^8)에서 표시된 특정 MDS행렬에 의한 곱으로 볼 수 있다.

2-5. AddRoundKey 단계

AddRoundKey단계에서, 서브키는 행렬과 결합된다. 각 라운드마다, 서브키는 Rijndael의 키 스케줄을 사용한 기본키에서 얻어진다. (각 서브키는 행렬의 크기와 같다.)

서브키는 각 행렬의 바이트와 해당 서브키의 바이트를 비트별 XOR로 결합하여 추가된다.

2-6. 암호의 최적화

32비트 이상의 단어를 사용하는 시스템에서는, Mixcolumns단계를 Table-lookup 방식으로 변환하여 SubBytes, ShiftRows 단계와 결합하여 암호의 실행 속도를 높일 수 있다.

여기에는 4개의 256-성분인 32비트 테이블이 필요하다. (총 4096바이트 차지) AddRoundKey 단계에서 라운드마다 16개의 Table-lookup방식과 1232비트의 XOR로 작동된다.

라운드마다 AddRoundKey 단계에서 4개의 32비트 XOR연산을 이용한 16개의 Table-lookup방식과 1232비트의 XOR로 작동된다. 대신에, Table-lookup 방식은 circular rotation 방식을 통해 단일 256-성분인 32비트 테이블(1024바이트 차지)로 수행될 수 있다.

바이트 지향 접근방식을 사용하면 SubBytes, ShiftRows 및 MixColumns 단계를 단일 라운드 작업으로 결합할 수 있습니다.

3. 안전성

NSA는 Rijndael을 포함한 AES최종 후보작들을 검토했고, 그 모든 것들이 미국 정부의 기밀이 아닌 정보를 지키는데 충분하다고 하였다. 2003년 6월, 미국 정부는 AES가 기밀정보를 보호하기 위해 사용될 수 있다고 발표했다.

AES알고리즘의 모든 키 길이(즉, 128, 192, 256)의 설계와 강도는 기밀수준의 정보를 보호하기 충분하다. TOP SECRET정보는 192 또는 256 키 길이를 사용해야 한다. 국가 보안 시스템 및 정보를 보호하기 위한 제품의 AES 구현은 획득 및 사용 전 NSA의 검토 및 인증을 받아야 한다.

AES는 128비트 키를 쓰는 10라운드, 192비트 키를 쓰는 12라운드, 256비트 키를 쓰는 14라운드가 있다.

2006년, 가장 잘 알려진 공격들은 128비트 키를 쓰는 7라운드, 192비트 키를 쓰는 8라운드, 256비트 키를 쓰는 9라운드이다.

3-1. 기지평문공격(Known attacks)

암호학자들에게, 암호학적 의미의 break(해독)은 Brute force attack(가능한 모든 키를 순차적으로 암호해독을 실행하는 것)보다 빠른 것을 의미한다. 따라서 break은 현재 기술로는 불가능한 결과를 포함할 수 있다. 실용적이지 않음에도, 이론적으로 break는 때때로 취약성 패턴에 대한 정보를 제공할 수 있다. 널리 알려진 블록 암호화 알고리즈에 대한 대중적으로 알려진 가장 성공적인 Brute force attack은 2006년 distributed.net의 64비트 RC5키이다.

키 길이보다 추가되는 비트마다 키 공간이 2배씩 증가하며, 키의 모든 값이 같다면, 이는 Brute force 키 검색 시간이 2배 증가한다는걸 의미한다. 이것은 Brute force 검색 시간이 키 길이에 따라 기하급수적으로 증가함을 의미한다. 키 길이 자체가 공격에 대한 안정성을 의미하지는 않는다. 매우 긴 키를 가진 암호들이 취약하다고 밝혀졌기 때문이다.

AES는 상당히 간단한 대수적 형태를 가지고 있다. 2002년, Nicolas Courtois와 Josef Pieprzyk는 비선형 의 낮은 복잡성으로 인해 AES 알고리즘의 약점이 있다고 주장하면서 “XSL 공격”이라고 칭한 공격법을 발표하였다. 그 이후, 다른 논문들은 원래 제시된 대로 그 공격이 실행 불가능하다는 것을 보여주었다.

AES 선별 과정에서, 경쟁했던 알고리즘 개발자들은 Rijndael 알고리즘에 대해”우리는 그것이 쓰이는 걸 걱정한다..보안이 중요한 프로그램에서”라고 말했다. 그러나 2000년 10월, 경쟁 알고리즘인 Twofish의 개발자인 Bruce Schneier는 Rijndael에 대한 성공적인 학문적 공격법이 언젠가 개발될 것이라고 생각했지만, 그는 “누군가 Rijndael통신을 읽을 수 있도록 하는 공격을 발견한다 믿지 않는다”라고 말했다.

2009년 5월까지, 모든 AES에 대한 성공적인 공개된 공격법은 일부 특정 구현체에 대한 부채널 공격 뿐이었다. 2009년, AES의 키 스케줄에 대한 단순성을 이용하고 2^119의 데이터 복잡도를 가진 새로운 연관 키 공격이 발견되었다. 2009년 12월에는 2^99.5로 향상되었다. 이것은 2009년 초  Alex Biryukov, Dmitry Khovratovich, Ivica Nikolić가 발견한 공격의 후속작으로, 2^35키 중 1개 키에서 2^96의 데이터 복잡도를 가지고 있다.그러나 적절하게 설계된 프로토콜(즉, 구현 소프트웨어)는 기본적으로 공격자가 관련성 있는 키를 선택하는 수단을 제한하여 연관 키를 허용하지 않도록 주의를 기울이기 때문에 연관 키 공격은 잘 설계된 암호 프로토콜에서는 문제가 되지 않는다.

또 다른 공격은 2009년 7월 30일 Bruce Schneier가 블로그에 올렸고, 2009년 8월 3일 프리프린트(출판 전 논문)로 출시되었다. Alex Biryukov, Orr Dunkelman, Nathan Keller, Dmitry Khovratovich, Adi Shamir가 새로 만든 이 공격은 9라운드에 256비트 키를 쓰는 버전

2개의 연관 키와 9라운드에 256비트 키를 쓰는 버전은 2^39의 시간복잡도가 쓰이는 AES-256을 공격한다.

Another attack was blogged by Bruce Schneier[[20]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard" \l "cite_note-27) on July 30, 2009, and released as a [preprint](https://en.wikipedia.org/wiki/Preprint)[[21]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-28) on August 3, 2009. This new attack, by Alex Biryukov, Orr Dunkelman, Nathan Keller, Dmitry Khovratovich, and [Adi Shamir](https://en.wikipedia.org/wiki/Adi_Shamir), is against AES-256 that uses only two related keys and 239 time to recover the complete 256-bit key of a 9-round version, or 245 time for a 10-round version with a stronger type of related subkey attack, or 270 time for an 11-round version. 256-bit AES uses 14 rounds, so these attacks are not effective against full AES.

The practicality of these attacks with stronger related keys has been criticized,[[22]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-29) for instance, by the paper on chosen-key-relations-in-the-middle attacks on AES-128 authored by Vincent Rijmen in 2010.[[23]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-30)

In November 2009, the first [known-key distinguishing attack](https://en.wikipedia.org/wiki/Known-key_distinguishing_attack) against a reduced 8-round version of AES-128 was released as a preprint.[[24]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-31) This known-key distinguishing attack is an improvement of the rebound, or the start-from-the-middle attack, against AES-like permutations, which view two consecutive rounds of permutation as the application of a so-called Super-S-box. It works on the 8-round version of AES-128, with a time complexity of 248, and a memory complexity of 232. 128-bit AES uses 10 rounds, so this attack is not effective against full AES-128.

The first [key-recovery attacks](https://en.wikipedia.org/wiki/Key-recovery_attack) on full AES were by Andrey Bogdanov, Dmitry Khovratovich, and Christian Rechberger, and were published in 2011.[[25]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-32) The attack is a [biclique attack](https://en.wikipedia.org/wiki/Biclique_attack) and is faster than brute force by a factor of about four. It requires 2126.2 operations to recover an AES-128 key. For AES-192 and AES-256, 2190.2 and 2254.6 operations are needed, respectively. This result has been further improved to 2126.0 for AES-128, 2189.9 for AES-192 and 2254.3 for AES-256,[[26]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-:0-33) which are the current best results in key recovery attack against AES.

This is a very small gain, as a 126-bit key (instead of 128-bits) would still take billions of years to brute force on current and foreseeable hardware. Also, the authors calculate the best attack using their technique on AES with a 128-bit key requires storing 288 bits of data. That works out to about 38 trillion terabytes of data, which is more than all the data stored on all the computers on the planet in 2016. As such, there are no practical implications on AES security.[[27]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-34) The space complexity has later been improved to 256 bits,[[26]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-:0-33) which is 9007 terabytes (while still keeping a time complexity of 2126.2).

According to the [Snowden documents](https://en.wikipedia.org/wiki/Edward_Snowden#Global_surveillance_disclosures), the NSA is doing research on whether a cryptographic attack based on [tau statistic](https://en.wikipedia.org/wiki/Kendall_tau_rank_correlation_coefficient) may help to break AES.[[28]](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard#cite_note-35)

At present, there is no known practical attack that would allow someone without knowledge of the key to read data encrypted by AES when correctly implemented.

3-2. 부채널 공격