AhnLab Cryptographic Module V1.1
보안정책서

KCMVP 암호모듈 검증

AhnLab

빈 페이지

<제목 차례>

1 장 암호모듈 명세

AhnLab Cryptographic Module V1.1은 변경 가능한 운영환경에서 동작하는 소프트웨어로 구성된 라이브러리 형태의 암호모듈로서, 운영체제의 유저 계층에서 동작하는 암호모듈이다.

[표 1-1] 암호모듈의 일반 사항

암호모듈명	AhnLab Cryptographic Module
버전	1.1
제조사(벤더)	안랩
전체 보안수준	보안수준 1
암호모듈 유형	소프트웨어 모듈
구성요소	단일 파일(동적 라이브러리)로 구성
운영환경	변경 가능한 운영환경, 유저계층 동작

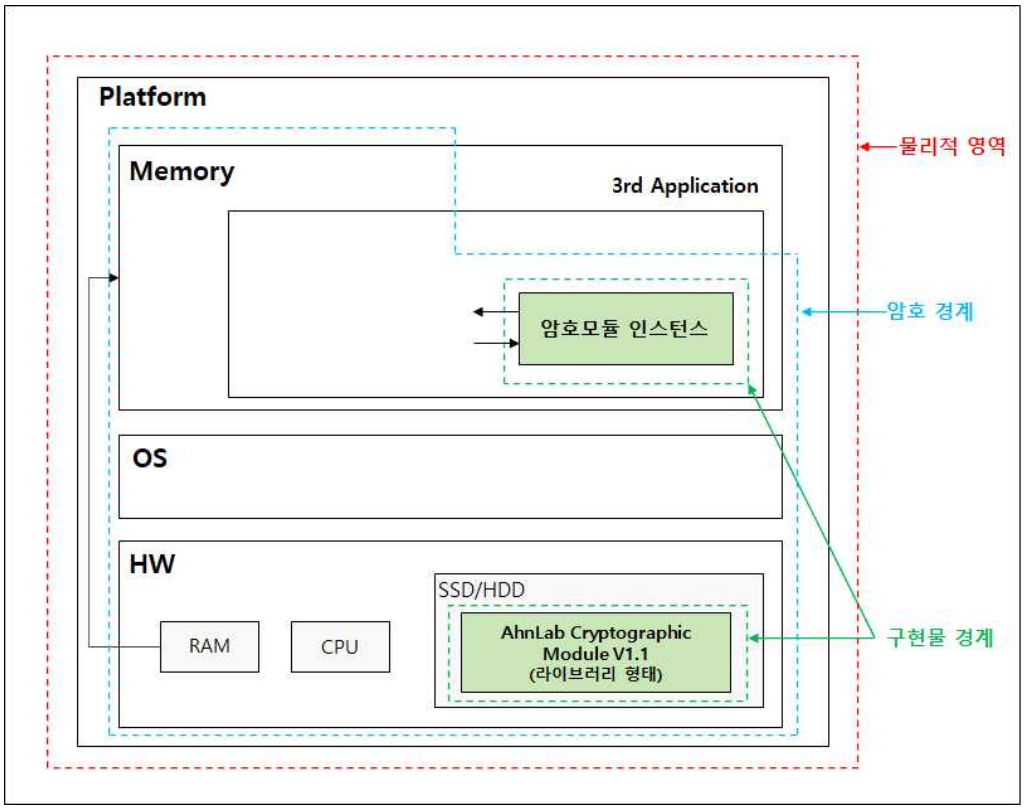
암호모듈에서 제공하는 모든 알고리즘의 구체적인 정보와 참조한 표준문서에 대한 정보는 [표 1-2]와 같다.

[표 1-2] 암호모듈의 암호알고리즘

구분				상세	참조표준
검증 대상	블록암호	SEED		|K| = 128비트 운영모드 : ECB, CBC 패딩방법 : PKCS, ZERO('0')	TTAS.KO-12.0004/R1(2005) ISO/IEC 18033-3(2010)
		ARIA		|K| = 128, 192, 256비트 운영모드 : ECB, CBC 패딩방법 : PKCS, ZERO('0')	KS X 1213-1(2014)
	해시	SHA-256			ISO/IEC 10118-3(2004) ISO/IEC 10118-3 Amd 1(2006)
		SHA-384			
		SHA-512			
	메시지 인증	HMAC	SHA-256		ISO/IEC 9797-2(2011)
			SHA-384		
			SHA-512		
	난수발생기	Hash_DRBG		해시 : SHA-256 논스 : 허용 안 함 개인화 문자열 : 허용 안 함 예측내성 : 비활성화 추가 입력 : 허용 리시딩 함수 : 제공 리시드 주기 : 2^{48}번	ISO/IEC 18031(2013) NIST SP 800-90A R1(2015)
	공개키 암호	RSAES		|n| = 2048비트 암호화 지수 = 65537 해시 : SHA-256	ISO/IEC 18033-2(2006) PKCS #1 v2.2(2012)
	전자서명	RSA-PSS		|n| = 2048비트 서명 검증 지수 = 65537 해시 : SHA-256	ISO/IEC 14888-2(2008) PKCS #1 v2.2(2012)
	키 설정	DH		(|P|, |Q|) = (2048, 256비트)	ISO/IEC 11770-3(2008)
비검증 대상	해당 사항 없음.				

1.1 암호경계

암호경계 내에는 소프트웨어 형태의 암호모듈만 위치한다. 암호모듈과 암호경계 외부 요소와의 입출력은 논리적 인터페이스를 통해 이루어지고, 물리적 포트를 이용한 입출력은 하지 않는다.



[그림 1-1] 암호경계

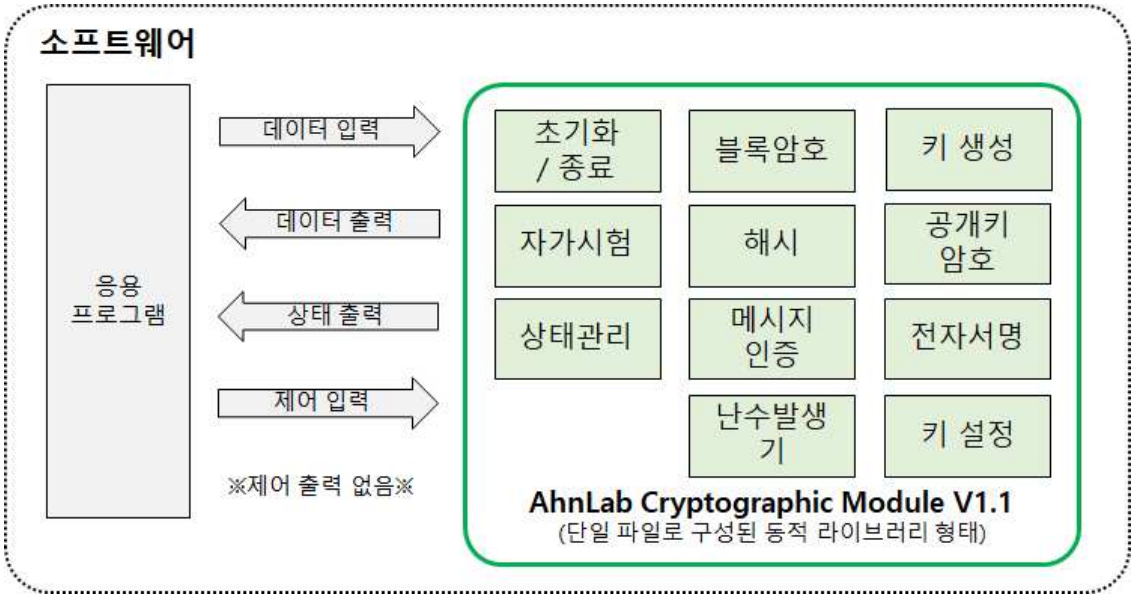
각 운영체제별 암호모듈의 구성요소는 다음과 같다. 암호모듈은 단일파일로 구성된 동적 라이브러리 형태의 소프트웨어 암호모듈이다.

[표 1-3] 암호모듈의 지원 운영체제

운영체제 구분			아키텍처	암호모듈 파일명
Windows	10	32비트	x86	libacm.dll
		64비트	arm64	libacm.dll
		64비트	x64	libacm.dll
	11	64비트		
ANOS (Customized Linux)	ANOS V2.7(Kernel 3.2)	64비트	x64	libacm.so
	ANOS V3.0(Kernel 4.4)	64비트	x64	
	ANOS V4.0(Kernel 5.10)	64비트	x64	

[그림1-2]은 암호모듈에 포함된 주요 기능과 외부 응용프로그램과의 논리적 인터페이스

연결을 보여주는 블록 다이어그램이다. 암호모듈의 모든 논리적 정보 흐름은 데이터 입력, 데이터 출력, 상태 출력, 제어 입력 인터페이스를 통해 이루어진다. 암호모듈은 단일 파일로 구성되었고, 다른 구성 요소가 없으므로 제어 출력 인터페이스를 가지지 않는다.



[그림 1-2] 암호모듈 블록 다이어그램

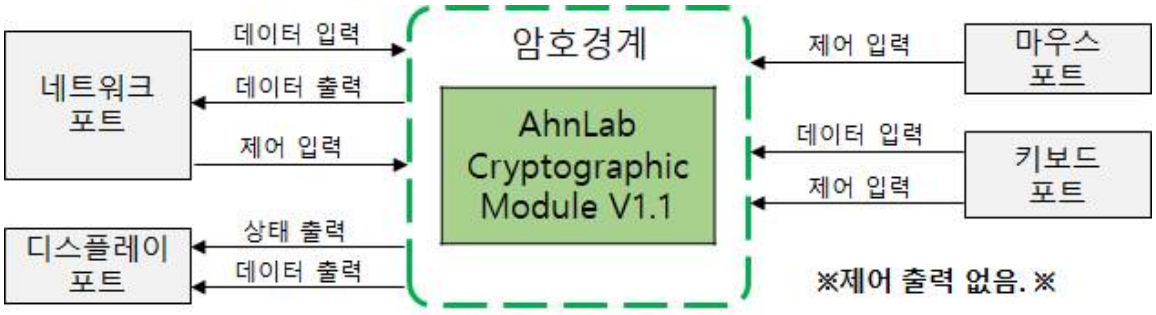
2 장 암호모듈 인터페이스

암호모듈의 논리적 인터페이스는 데이터 입력, 데이터 출력, 제어입력, 상태출력과 같이 4개의 인터페이스가 존재하며, 제어출력 인터페이스는 존재하지 않는다.

[표 2-1] 논리적 인터페이스와 물리적 포트의 관계

논리적 인터페이스	논리적 정보	물리적 포트
데이터 입력	API에 입력되는 매개변수 데이터	네트워크, 키보드
데이터 출력	API에서 출력되는 매개변수 데이터	네트워크, 디스플레이
제어 입력	API 호출	네트워크, 키보드, 마우스
제어출력	해당 사항 없음.	
상태 출력	API 실행결과(오류코드), 암호모듈의 현재 상태	디스플레이

암호모듈에 입출력되는 모든 논리적 정보의 흐름은 논리적 인터페이스와 물리적 포트에 제한되며, 특정 논리적 인터페이스는 여러 개의 물리적 포트를 공유한다. 예를 들면 제어 입력 인터페이스는 네트워크/마우스/키보드 포트와 연결된다.



[그림 2-1] 물리적 포트와 논리적 인터페이스의 관계에 대한 블록도

3 장 역할, 서비스 및 인증

3.1 역할

암호모듈은 다음과 같은 운영자의 역할을 지원하며, 복수의 운영자를 지원하지 않는다.

◈ 암호 관리자 : 암호관리자는 암호모듈의 설치/삭제, 초기화/종료, 자가시험과 같은 관리를 담당한다. 암호관리자는 배포 받은 암호모듈의 ***정합성 시험***을 반드시 실시해야 한다.

◈ 암호 사용자 : 사용자는 초기화된 암호모듈의 모든 암호서비스를 이용할 수 있다. 또한 자가시험을 실시할 수 있다.

3.2 서비스

서비스는 모듈에서 수행되는 모든 동작, 서비스, 기능으로 정의된다. 암호모듈은 [표 3-1]과 같이 운영자에게 서비스를 제공하며, 비보안 관련 서비스는 제공하지 않는다.

[표 3-1] 제공 서비스

서비스		역할	제어 입력	기능 설명
초기화		관리자	ACM_Initialize	암호모듈의 초기화 관련 작업을 한다. 초기화가 완료되면, 사용자에게 핸들이 발급된다. 사용자는 이 핸들을 소유하고 있어야만 암호모듈에서 제공되는 암호서비스를 이용할 수 있다.
종료		관리자	ACM_Finalize	암호모듈을 안전하게 종료한다.
상태표시		관리자	ACM_GetState	암호모듈의 현재 상태를 출력한다.
모듈 정보 조회		관리자	ACM_GetACMInfo	암호모듈의 정보(이름, 버전, 해시값)를 조회한다.
자가시험		사용자	ACM_SelfTest	암호모듈에서 제공되는 모든 암호알고리즘에 대해서 정상적으로 동작하는지 시험한다
블록암호	암호화	사용자	ACM_SYMM_EncryptInit ACM_SYMM_Encrypt, ACM_SYMM_EncryptUpdate, ACM_SYMM_EncryptFinal,	블록암호 알고리즘을 사용해서 초기화, 암호화, 부분 업데이트, 작업(패딩 처리 등)한다.
	복호화	사용자	ACM_SYMM_DecryptInit, ACM_SYMM_Decrypt,	블록암호 알고리즘을 사용해서 초기화, 복호화, 부분 업데이트, 작업(패딩 처리

			ACM_SYMM_DecryptUpdate, ACM_SYMM_DecryptFinal,	등)한다.
공개키 암호	암호화	사용자	ACM_ASYMM_Encrypt	개키 암호알고리즘을 이용해서 평문을 공개키로 암호화한다.
	복호화	사용자	ACM_ASYMM_Decrypt	공개키 암호알고리즘을 이용해서 암호문을 개인키로 복호화한다.
해시함수		사용자	ACM_MD_DigestInit, ACM_MD_Digest, ACM_MD_DigestUpdate, ACM_MD_DigestFinal	해시 연산을 수행하기 위한 초기화, 해시계산, 부분 압축 등 작업을 한다.
메시지 인증		사용자	ACM_MAC_DigestInit, ACM_MAC_Digest, ACM_MAC_DigestUpdate, ACM_MAC_DigestFinal	HMAC 알고리즘을 수행하기 위한 초기화, MAC 계산, 부분 압축 등 작업을 한다.
전자서명		서명 생성	ACM_DS_SignInit, ACM_DS_Sign, ACM_DS_SignUpdate, ACM_DS_SignFinal	전자서명 알고리즘을 이용해서 서명을 수행하기 위해, 초기화, 개인키로 서명 , 부분 압축 등 작업을 한다.
		서명 검증	ACM_DS_VerifyInit, ACM_DS_Verify, ACM_DS_VerifyUpdate, ACM_DS_VerifyFinal	전자서명 알고리즘을 이용해서 서명 검증을 수행하기 위해, 초기화, 공개키로 검증, 부분 압축 등 작업을 한다.
리시딩		사용자	ACM_KM_ReseedRBG	새로운 엔트로피 입력으로 난수발생기를 리시딩한다.
난수 생성		사용자	ACM_KM_GenerateRand	사용자가 요구한 크기만큼 난수열을 생성한다.
비밀키 생성		사용자	ACM_KM_GenerateKey	사용자가 요구한 크기만큼 비밀키를 생성한다.
공개키 쌍 생성		사용자	ACM_KM_GenerateKeyPair	RSA, DH에 사용되는 암호키 쌍(공개키, 개인키)을 생성한다.
키 설정		사용자	ACM_KM_KeyAgreement	키 설정 알고리즘을 이용해서 양자간 비밀 키를 설정한다.

4 장 소프트웨어/펌웨어 보안

암호모듈은 초기화 시 무결성 시험을 한다. 무결성 시험에서 사용되는 알고리즘은 HMAC with SHA-512이다. 무결성 시험에서 실패하면, 암호모듈의 상태는 치명적 오류 상태로 천이된다.

5 장 운영환경

5.1 실행 하드웨어 환경

암호모듈을 실행하기 위한 하드웨어 요구사양은 다음과 같다.

[표 5-1] 암호모듈 실행 하드웨어 요구사양

구분	최소사양	권장사양
CPU	Pentium 233MHz	PAE 지원 Dual Core
메모리	64MB	1G
저장공간	50MB 여유 공간	50MB 여유 공간

5.2 지원 운영체제

암호모듈은 다음과 같은 운영체제에서 동작한다.

[표 5-2] 암호모듈의 지원 운영체제

운영체제 구분			아키텍처	암호모듈 파일명
Windows	10	32비트	x86	libacm.dll
		64비트	arm64	libacm.dll
		64비트	x64	libacm.dll
	11	64비트		
ANOS (Customized Linux)	ANOS V2.7(Kernel 3.2)	64비트	x64	libacm.so
	ANOS V3.0(Kernel 4.4)	64비트	x64	
	ANOS V4.0(Kernel 5.10)	64비트	x64	

5.3 QRNG(Quantum Random Number Generator) 하드웨어 정보

암호모듈은 QRNG 하드웨어가 장착된 Linux 운영환경에서 QRNG의 출력을 잡음원으로 추가 사용한다.

[표 5-3] QRNG 하드웨어 정보

개발사	㈜이와이엘 (www.eylpartners.com)	
QRNG 하드웨어 타입	USB	PCI-Express
모델번호	Q02UG1	Q02PG4

6 장 물리적 보안

암호모듈은 소프트웨어 라이브러리 형태의 모듈이기 때문에 물리적 보안 기능이 제공되지 않는다.

7 장 비침투 보안

암호모듈은 소프트웨어 라이브러리 형태의 모듈이기 때문에 비침투 보안 기능이 제공되지 않는다.

8 장 중요 보안 매개변수 관리

암호모듈이 관리하는 SSP는 [표 8-1]과 같다.

[표 8-1] 중요보안매개변수

구분			크기	생성	설정	주입	출력	저장	제로화
핵심보안매개변수(CSP)	비밀키	SEED 비밀키	SEED-128:|K|=16바이트	○	×	×	×	×	○
		SEED 라운드키	SEED-128:|RK|=128바이트	○	×	×	×	×	○
		ARIA 비밀키	ARIA-128:|K| = 16바이트 ARIA-192:|K| = 24바이트 ARIA-256:|K| = 32바이트	○	×	×	×	×	○
		ARIA 라운드키	ARIA-128:|RK| = 208바이트 ARIA-192:|RK| = 240바이트 ARIA-192:|RK| = 272바이트	○	×	×	×	×	○
		HMAC 비밀키	HMAC-SHA-256:|K| ≥32바이트 HMAC-SHA-384:|K| ≥48바이트 HMAC-SHA-512:|K| ≥64바이트	○	×	×	×	×	○
	개인키	RSAES 개인키	|n|=2048비트 일 경우, |d|≥1024비트	○	×	×	×	×	○
		RSA-PSS 개인키	|n|=2048비트 일 경우, |s|≥1024비트	○	×	×	×	×	○
		DH 개인키	|r|≥256비트	○	×	×	×	×	○
	컨디셔닝 처리된 엔트로피 입력		32바이트	○	×	×	×	×	○
	난수발생기 내부상태	난수발생기 내부상태 = {V, C}	|V|+|C|=55+55=110	○	×	×	×	×	○
	마스크	핸들 보호 마스크	32비트 운영체제 : 32비트 64비트 운영체제 : 64	○	×	×	×	×	×

			비트						
	민감한 내부 지역변수	암호모듈 내부연산에 사용되는 변수 중 민감한 정보		○	×	×	×	×	○
	공유키	DH로 합의된 공유키	2048비트	×	○	×	×	×	○
공개보안 매개변수 (PSP)	핸들	초기화 시 사용자에게 발급된 핸들	32비트 운영체제 : 32비트 64비트 운영체제 : 64비트						
	공개키	RSAES 공개키	|n| = 2048비트	○	×	×	×	×	○
		RSA-PSS 공개키	|n| = 2048비트	○	×	×	×	×	○
		DH 공개키	|P| = 2048비트 |Q| = 256비트	○	×	×	×	×	○

9 장 자가시험

암호모듈은 정상적으로 동작할 수 있는지 확인하기 위해 사용자의 개입 없이 스스로 자가시험을 진행한다. 자가시험에서 실패 시, 암호모듈은 심각한 오류상태로 천이된다. 이 때는 종료를 제외한 어떤 API도 실행할 수 없으며, 사용자는 종료 API를 호출해서 암호모듈을 종료해야 한다. 자가시험은 [표 9-1]과 같이 동작 전 자가시험, 조건부 자가시험 그리고 사용자 자가시험으로 구분된다.

[표 9-1] 자가시험

자가시험 구분	시험 내용
동작 전 자가시험	암호알고리즘 시험
	난수발생기 잡음원 건전성 시험
	소프트웨어 무결성 시험
조건부 자가시험	암호알고리즘 시험
	난수발생기 잡음원 건전성 시험
	암호키 쌍 일치 시험
사용자 자가시험	암호알고리즘 시험
	난수발생기 잡음원 건전성 시험
	소프트웨어 무결성 시험

10 장 생명주기 보증

10.1 배포 및 운영

암호관리자는 암호모듈 구성요소의 **정합성 시험**을 반드시 해야 한다.

10.2 암호모듈 폐기

암호모듈을 사용 종료 및 파기하기 위해서는 관리자가 저장장치에서 암호모듈을 삭제해야 한다.

10.3 안내서

암호모듈의 설치 및 사용 방법은 다음과 같은 문서에 상세히 기술되어 있다.

◈ 제품 사용 설명서 : 암호모듈의 설치 방법, 간단한 사용 예제를 설명한 문서

◈ 함수 설명서 : 사용자 API, 상수, 구조체 등을 설명한 문서

11 장 기타 공격에 대한 대응

암호모듈은 정적분석 도구를 이용해서 암호모듈를 검토하고 대응하였다. 또한 시간 및 전력분석에 대응하기 위해서 멱승 연산 시 고정 길이 윈도우를 사용한다.