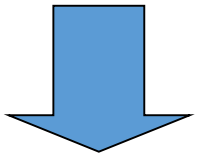
# 암호화와 인증서

# 1. 암호화의 역할

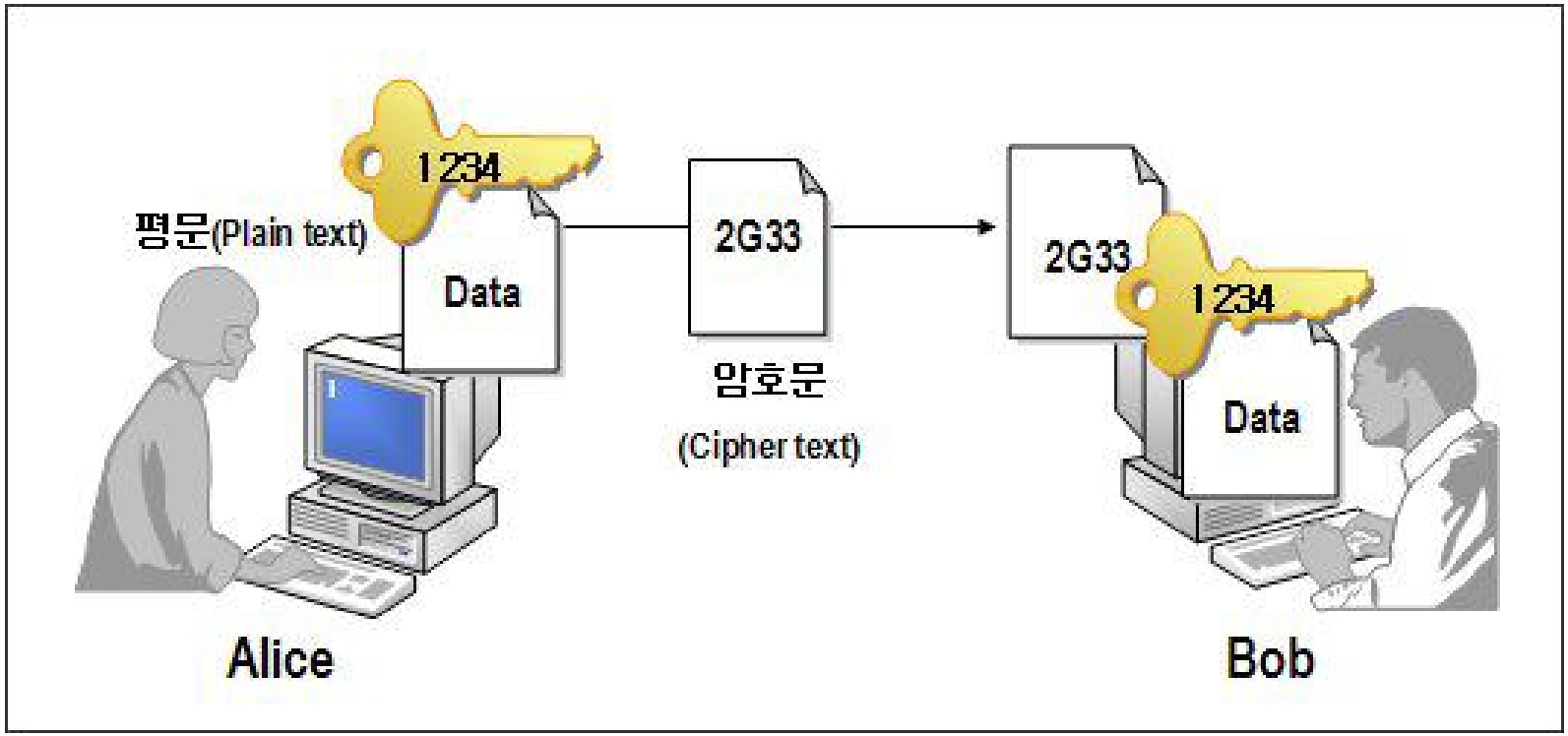
- Confidentiality (기밀성)
- Authenticity (신뢰성)
- Integrity (무결성)



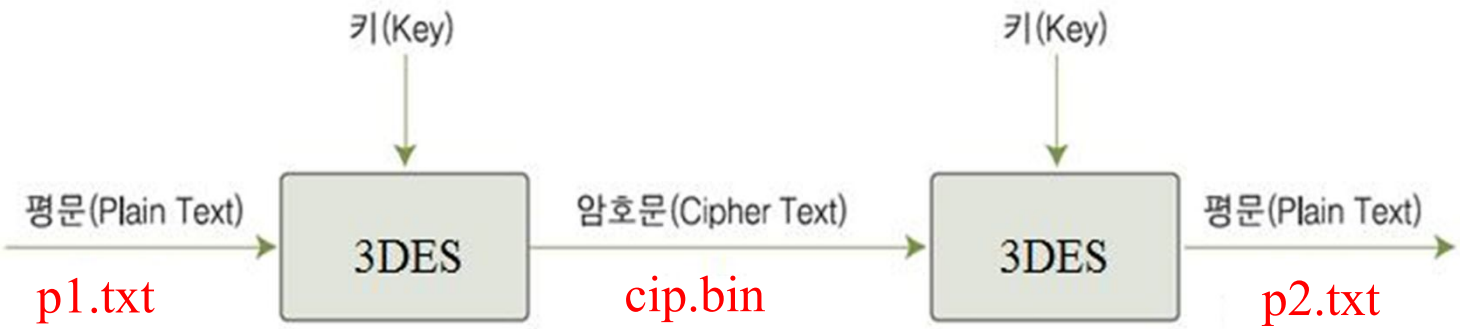
Key를 이용한 암호화 기술로서 해결

# 1) Symmetric Key

- Encryption Key = Decryption Key




  - DES, 3DES 등의 프로토콜이 해당됨

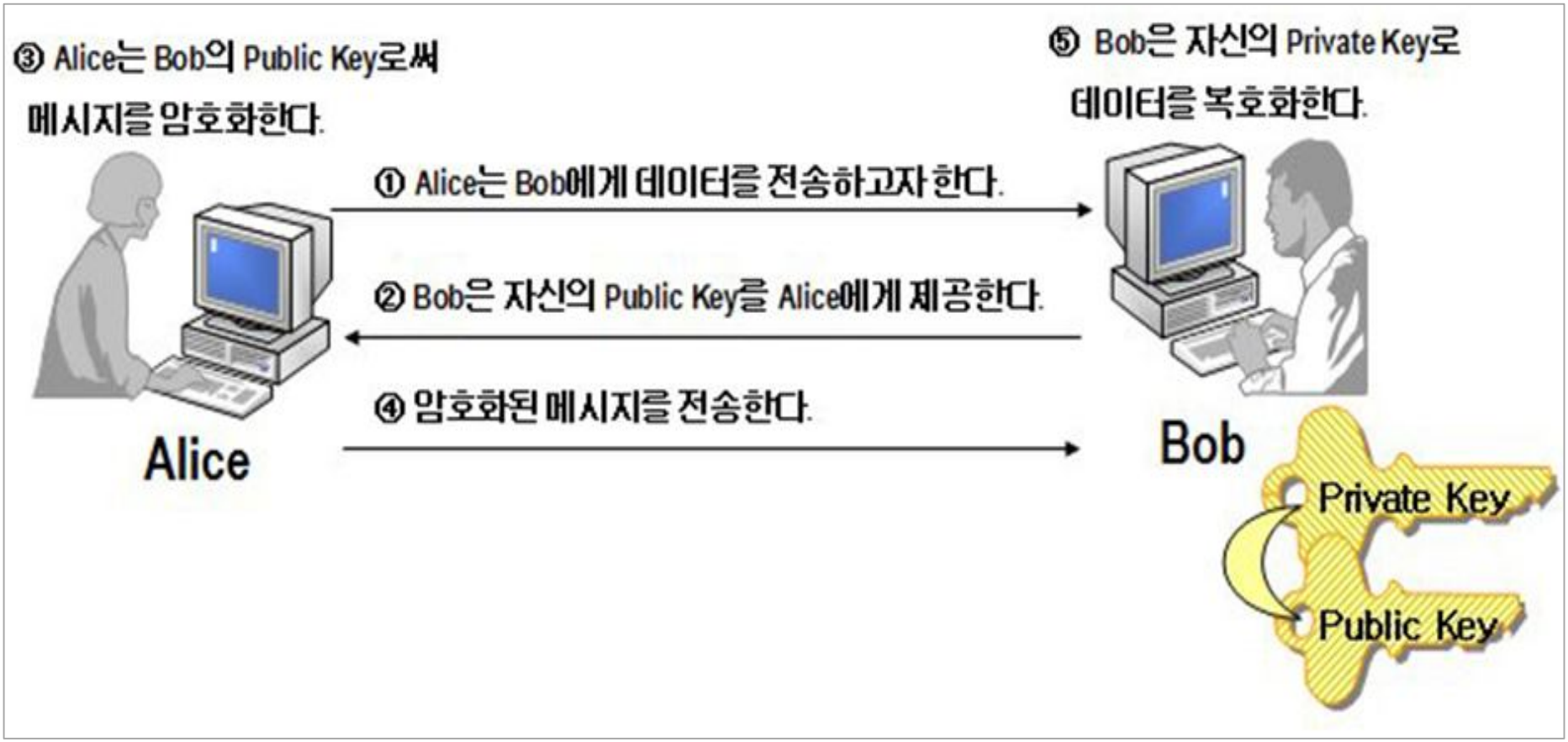


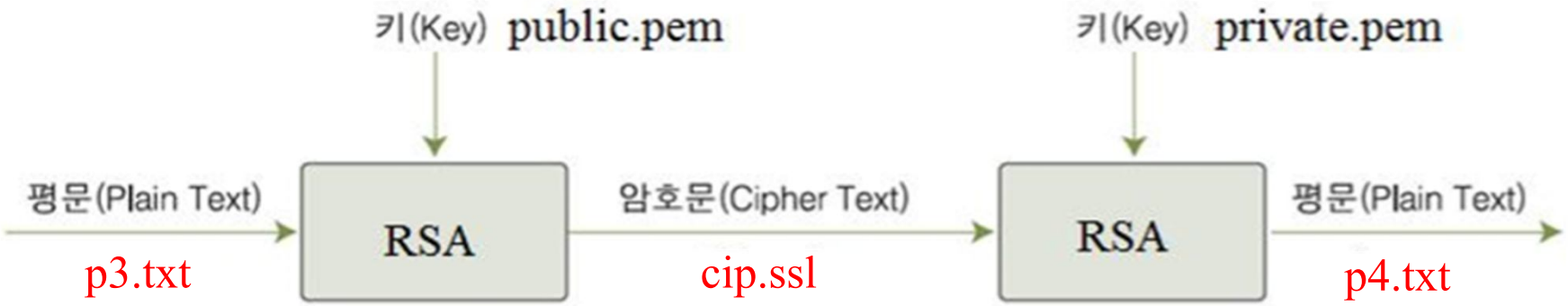

| | |
|---|---|
| 암호화 | echo ‘This is the plain test' > p1.txt<br><br>openssl enc -e -des3 -salt -in p1.txt -out cip.bin |
| 복호화 | openssl enc -d -des3 -in cip.bin -out p2.txt |

# 2) Public Key

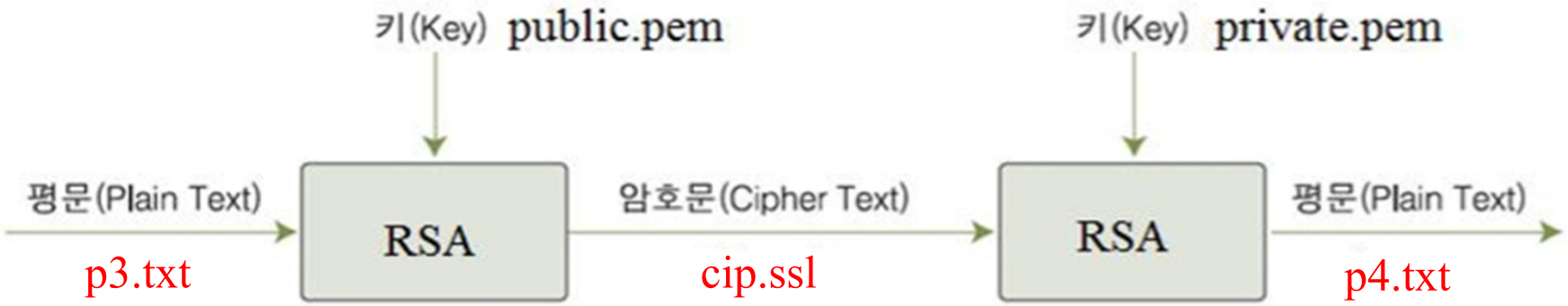
- Encryption Key ≠ Decryption Key
    - 비대칭키, 공용키, 페어(Pair)키에 해당함
- Public Key와 Private Key로 구성
    - Public Key(공용키) = 공개가 되는 키
    - Private Key(개인키) = 오직 발행주체만이 가지는 비밀키
- RSA 프로토콜

# ① Public Key 암호화(Confidentiality 제공)



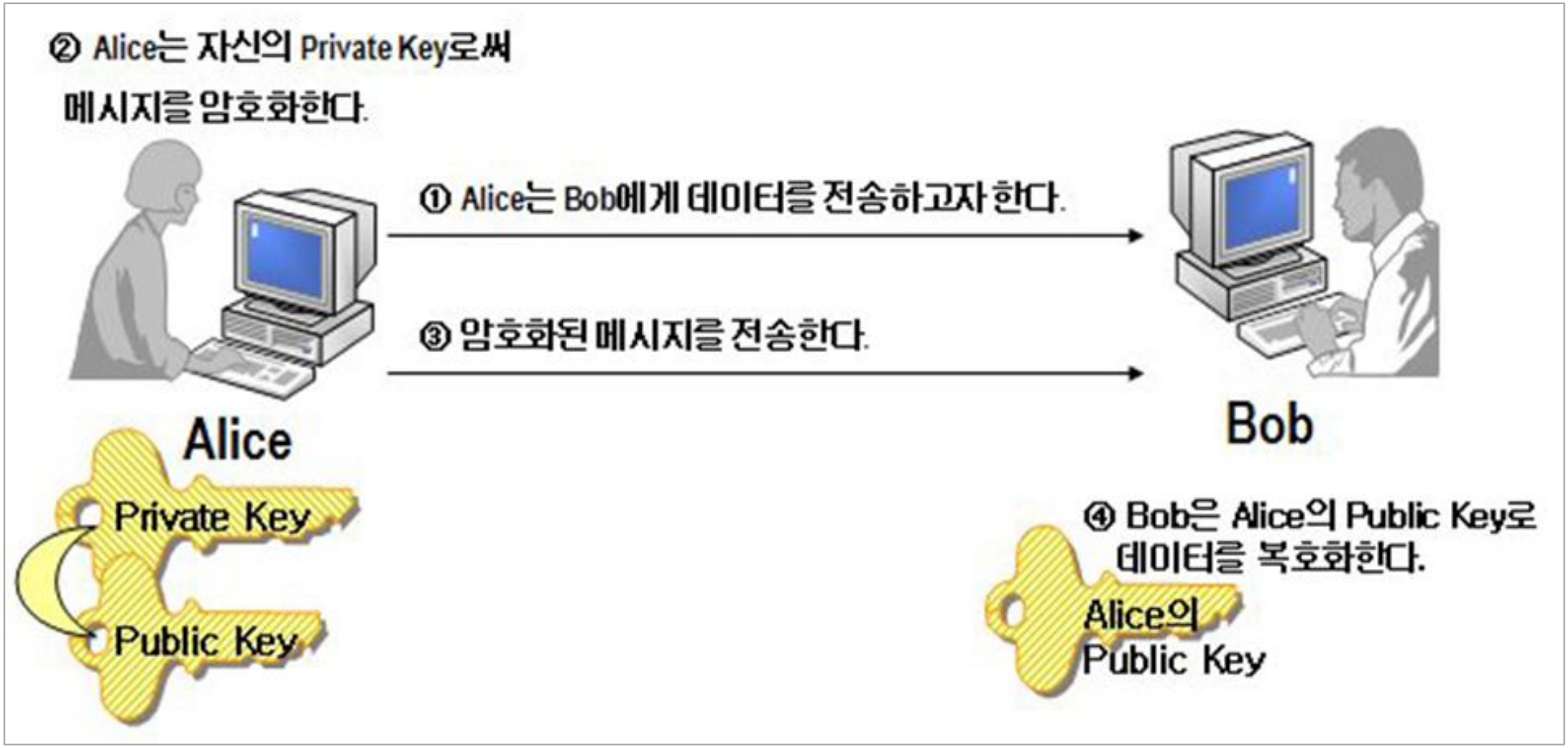



| | |
|---|---|
| 개인키 생성 | openssl genrsa -out private.pem 2048 |
| 공개키 생성 | openssl rsa -in private.pem -out public.pem -outform PEM -pubout |

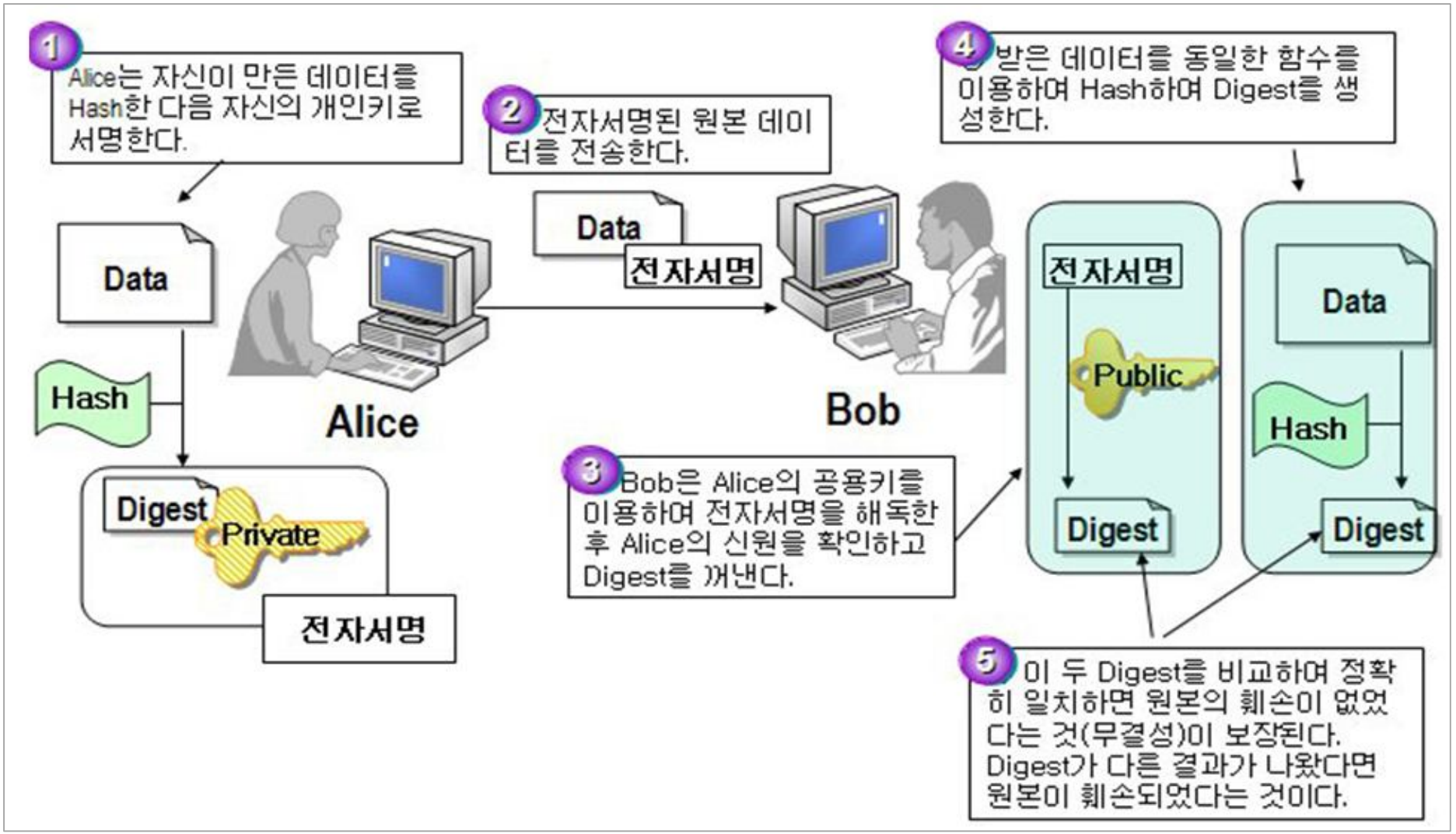



| | |
|---|---|
| 공개키로 암호화 | echo 'This is encryption text' > p3.txt<br>openssl pkeyutl -encrypt -inkey public.pem -pubin -in p3.txt -out cip.ssl |
| 개인키로 복호화 | openssl pkeyutl -decrypt -inkey private.pem -in cip.ssl -out p4.txt |

# ② Private Key 암호화(Authenticity 제공)

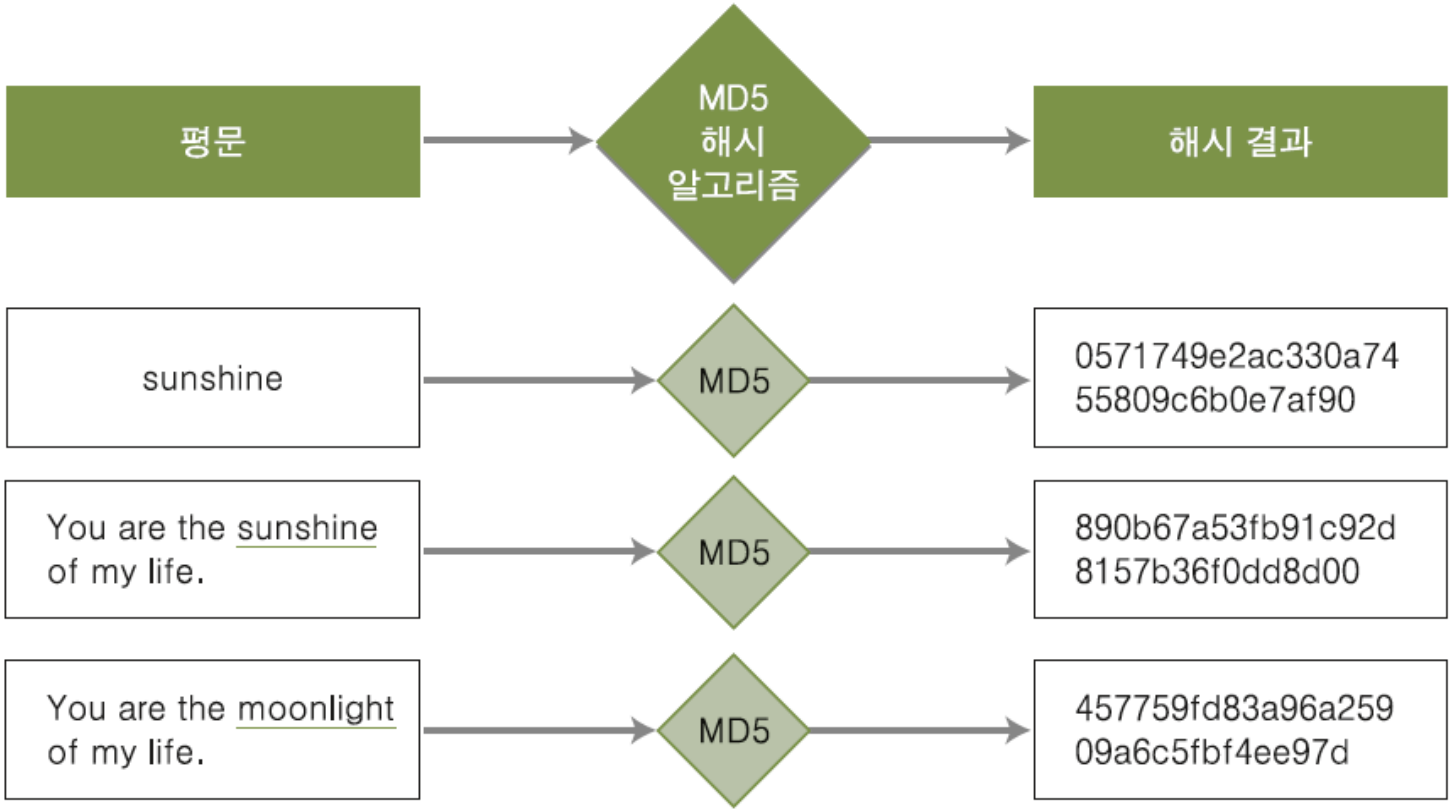


# ③ Hash Function & Digest(Integrity 제공)



# 3) Hash Algorithm

- 해시 : 하나의 문자열을, 이를 상징하는 더 짧은 길이의 값이나 키로 변환하는 것



- MD 알고리즘

  - MD2, MD4, MD5 이렇게 세 가지가 있음.

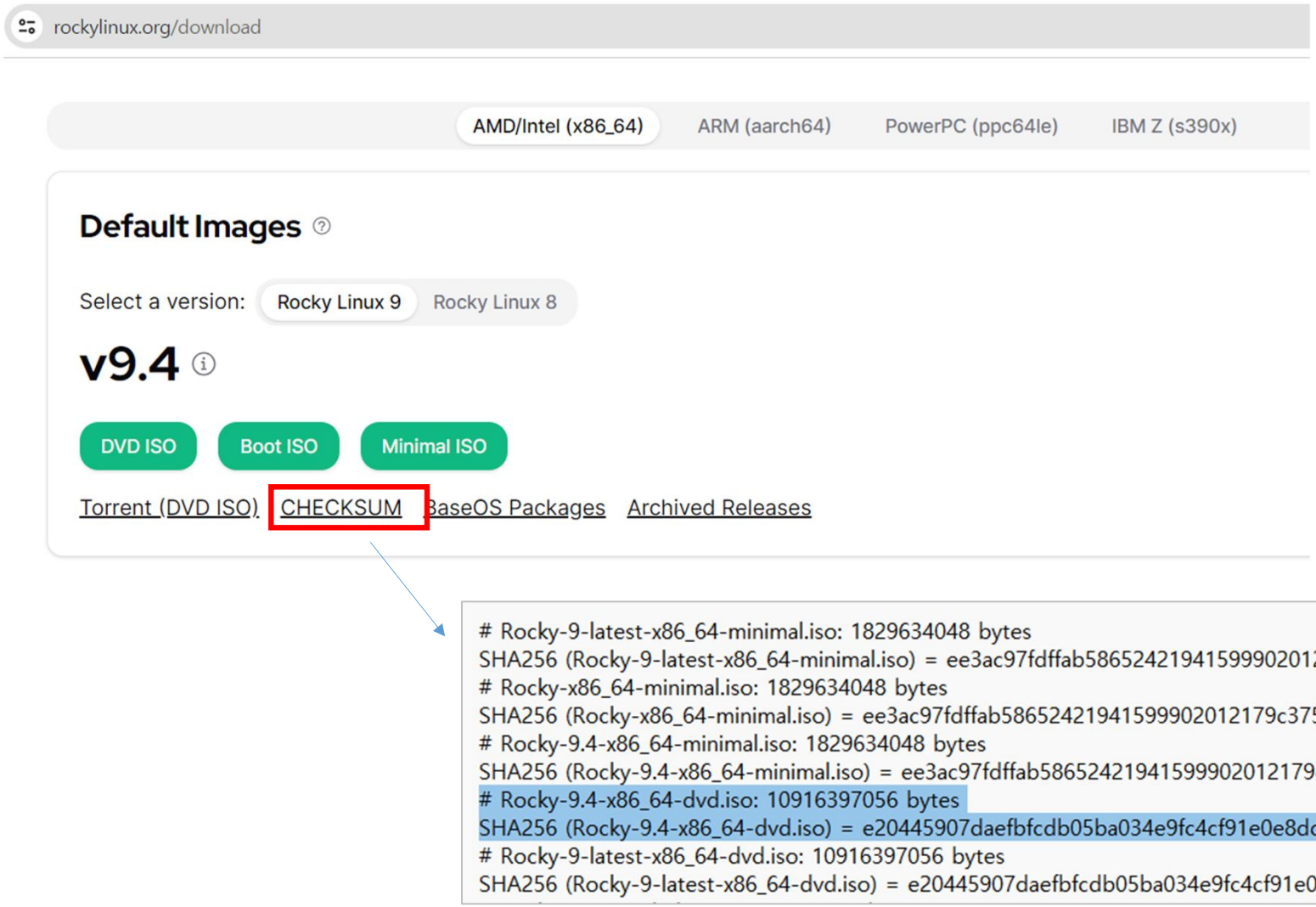
  - MD5 알고리즘은 MD4의 확장판으로, MD4보다 속도가 빠르지는 않지만 데이터 보안성에 있어 더 많은 확신을 제공

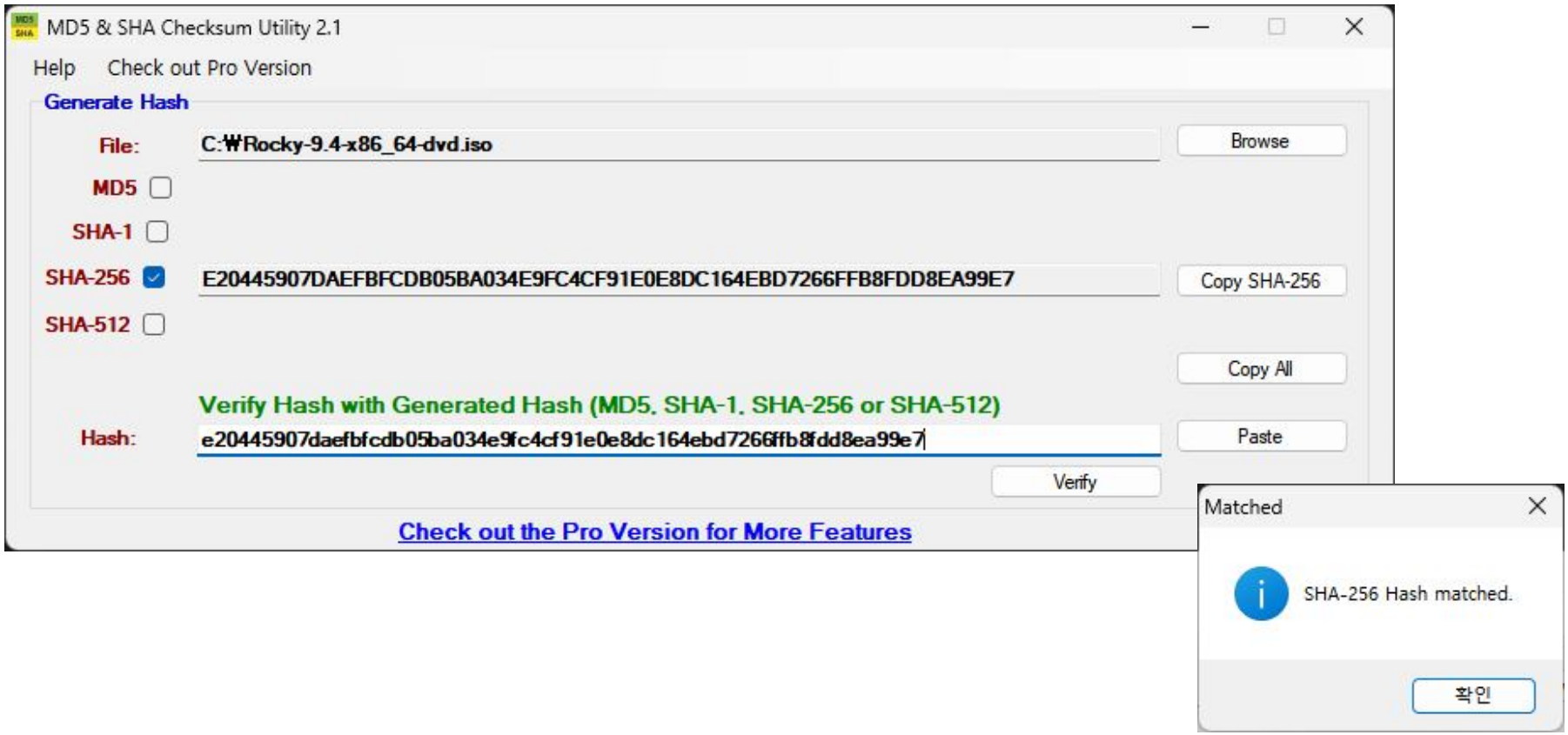
- SHA 알고리즘

  - 160비트의 값을 생성하는 해시 함수로, MD4가 발전한 형태

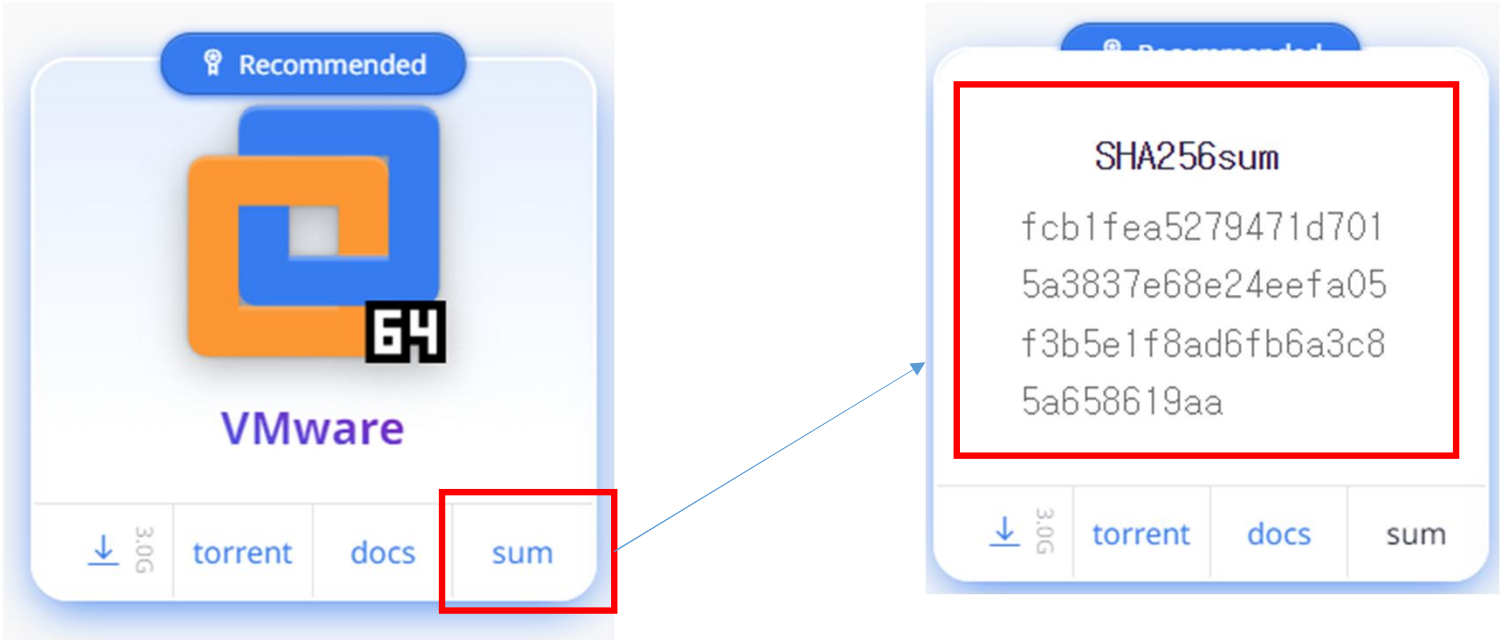
  - MD5보다 조금 느리지만 좀더 안전한 것으로 알려져 있음

| 알고리즘 | 블록 크기 | 해시 결과값 길이 | 해시 강도 |
|---|---|---|---|
| SHA-1 | 512비트 | 160비트 | 0.625 |
| SHA-256 | 512비트 | 256비트 | 1 |
| SHA-384 | 1024비트 | 384비트 | 1.5 |
| SHA-512 | 1024비트 | 512비트 | 2 |


rockylinux.org/download
AMD/Intel (x86_64)
ARM (aarch64)
PowerPC (ppc64le)
IBM Z (s390x)
Default Images
Select a version:
Rocky Linux 9
Rocky Linux 8
v9.4
DVD ISO
Boot ISO
Minimal ISO
Torrent (DVD ISO)
CHECKSUM
BaseOS Packages
Archived Releases
# Rocky-9-latest-x86_64-minimal.iso: 1829634048 bytes
SHA256 (Rocky-9-latest-x86_64-minimal.iso) = ee3ac97fdffab5865242194159990201 2179c37535aece76824673105169c4a2
# Rocky-x86_64-minimal.iso: 1829634048 bytes
SHA256 (Rocky-x86_64-minimal.iso) = ee3ac97fdffab5865242194159990201 2179c37535aece76824673105169c4a2
# Rocky-9.4-x86_64-minimal.iso: 1829634048 bytes
SHA256 (Rocky-9.4-x86_64-minimal.iso) = ee3ac97fdffab5865242194159990201 2179c37535aece76824673105169c4a2
# Rocky-9.4-x86_64-dvd.iso: 10916397056 bytes
SHA256 (Rocky-9.4-x86_64-dvd.iso) = e20445907daefbfcdb05ba034e9fc4cf91e0e8dc164ebd7266ffb8fdd8ea99e7
# Rocky-9-latest-x86_64-dvd.iso: 10916397056 bytes
SHA256 (Rocky-9-latest-x86_64-dvd.iso) = e20445907daefbfcdb05ba034e9fc4cf91e0e8dc164ebd7266ffb8fdd8ea99e7


MD5 & SHA Checksum Utility 2.1
Help
Check out Pro Version
Generate Hash
File:
C:₩Rocky-9.4-x86_64-dvd.iso
Browse
MD5
SHA-1
SHA-256
E20445907DAEFBFCDB05BA034E9FC4CF91E0E8DC164EBD7266FFB8FDD8EA99E7
Copy SHA-256
SHA-512
Copy All
Verify Hash with Generated Hash (MD5, SHA-1, SHA-256 or SHA-512)
Hash:
e20445907daefbfcdb05ba034e9fc4cf91e0e8dc164ebd7266ffb8fdd8ea99e7
Paste
Verify
Check out the Pro Version for More Features
Matched
SHA-256 Hash matched.
확인

- https://www.kali.org/get-kali/#kali-virtual-machines




- https://ftp.kaist.ac.kr/CentOS/7.9.2009/isos/x86_64/

# 2. SSL/TLS

· 응용계층을 보호하는 프로토콜

· 웹 브라우저에 기본적으로 탑재되어 있음

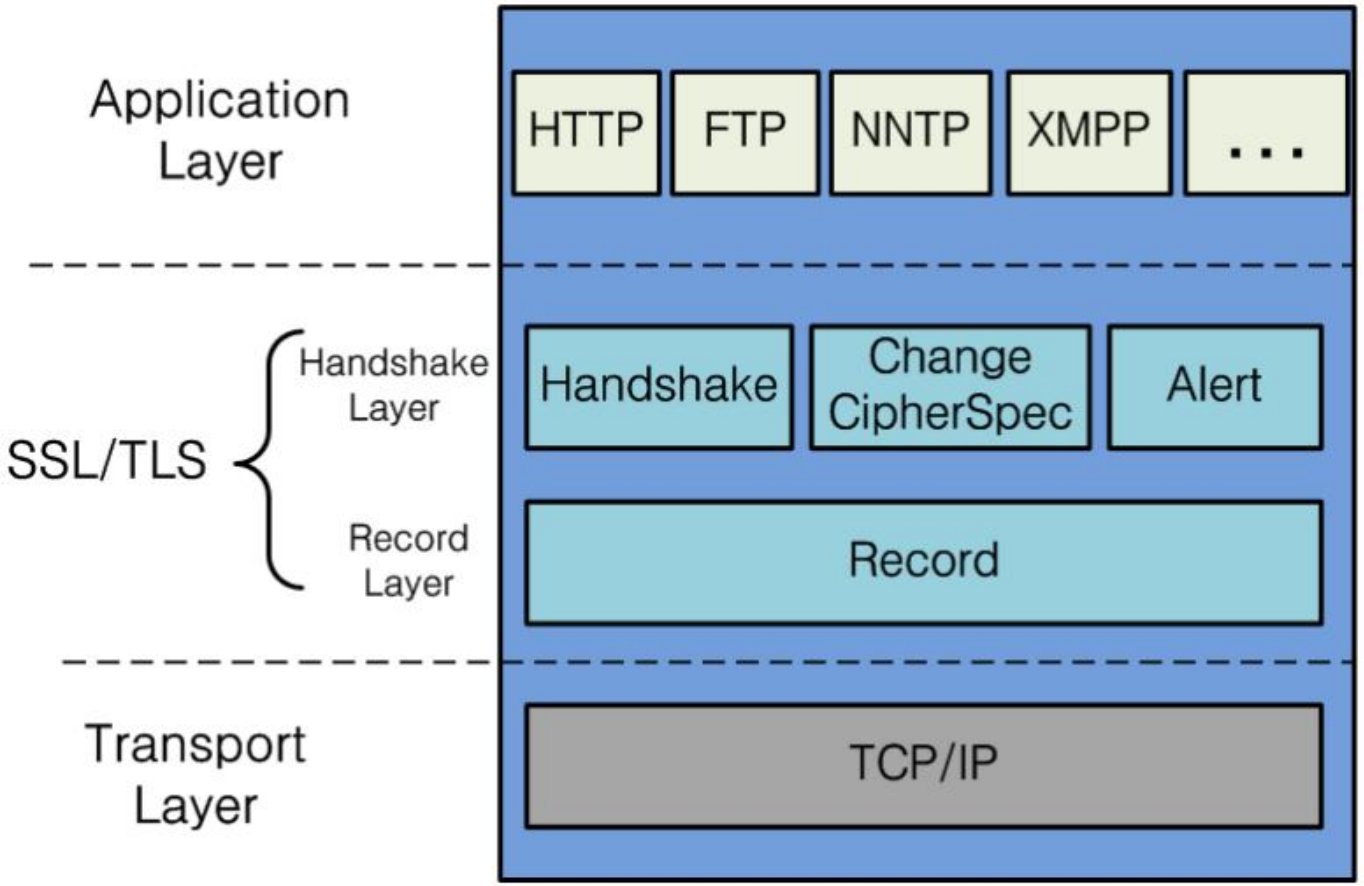
· IETF에서 TLS 1.0 발표,  SSL 3.1 이라고도 함

· TLS 프로토콜은 SSL 프로토콜을 사용할 수 있도록 구성

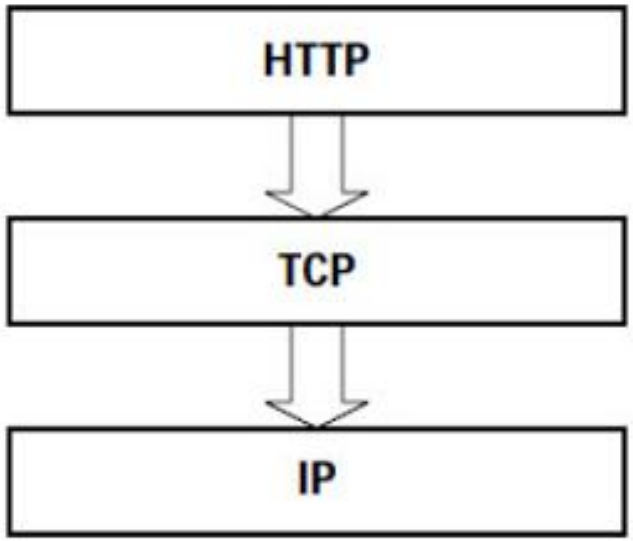
· SSL/TLS 프로토콜이라는 용어 사용

| SSL(Secure Socket Layer) | TLS(Transport Layer Security) |
|---|---|
| 응용계층을 보호하는 프로토콜 | 전송 계층 상위에서 동작하여 응용계층 암호화<br>HTTP 패킷 보호가 주용도 |

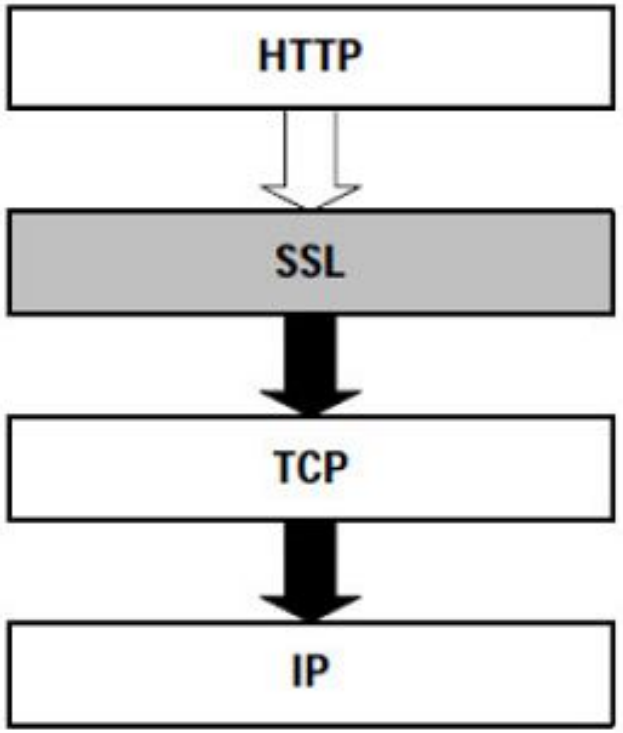
# 1) SSL/TLS Protocol stack



- HTTP와 HTTPS

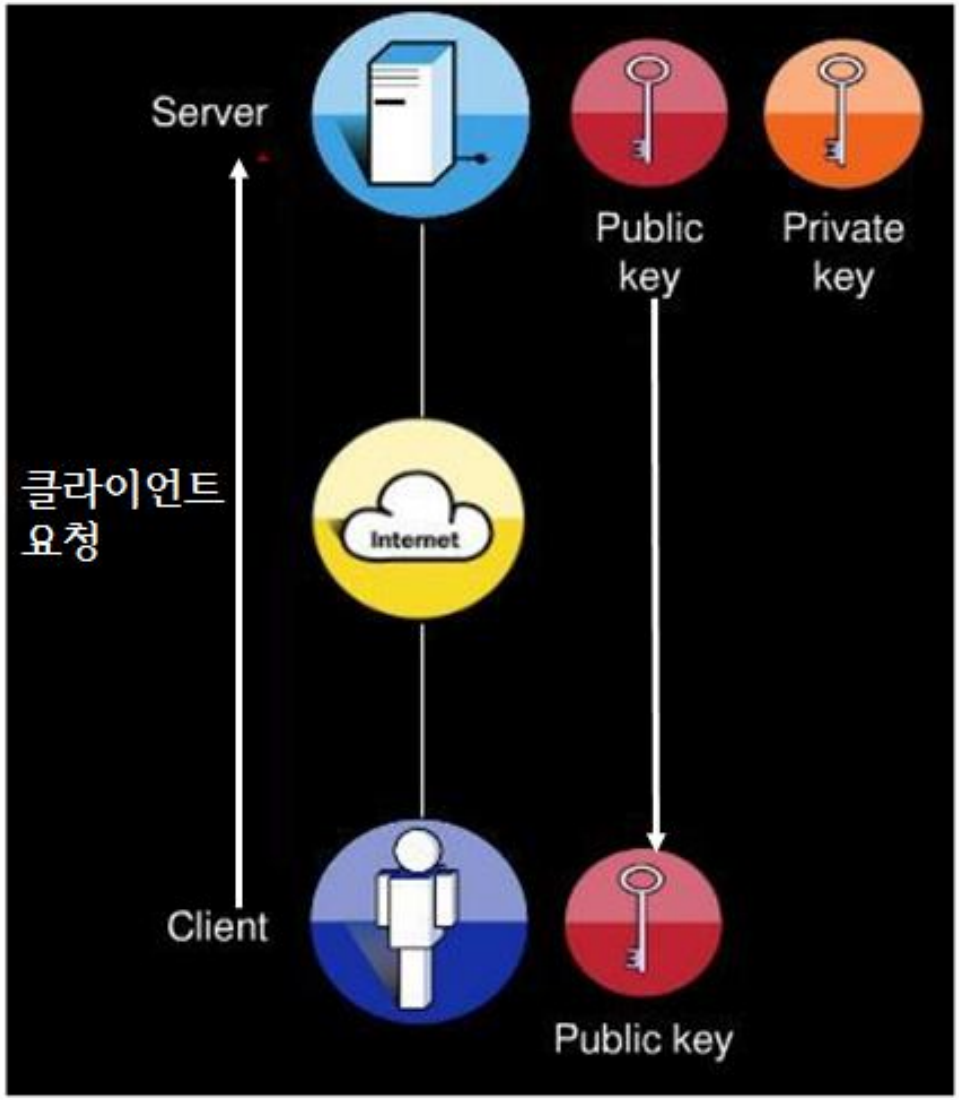



HTTP

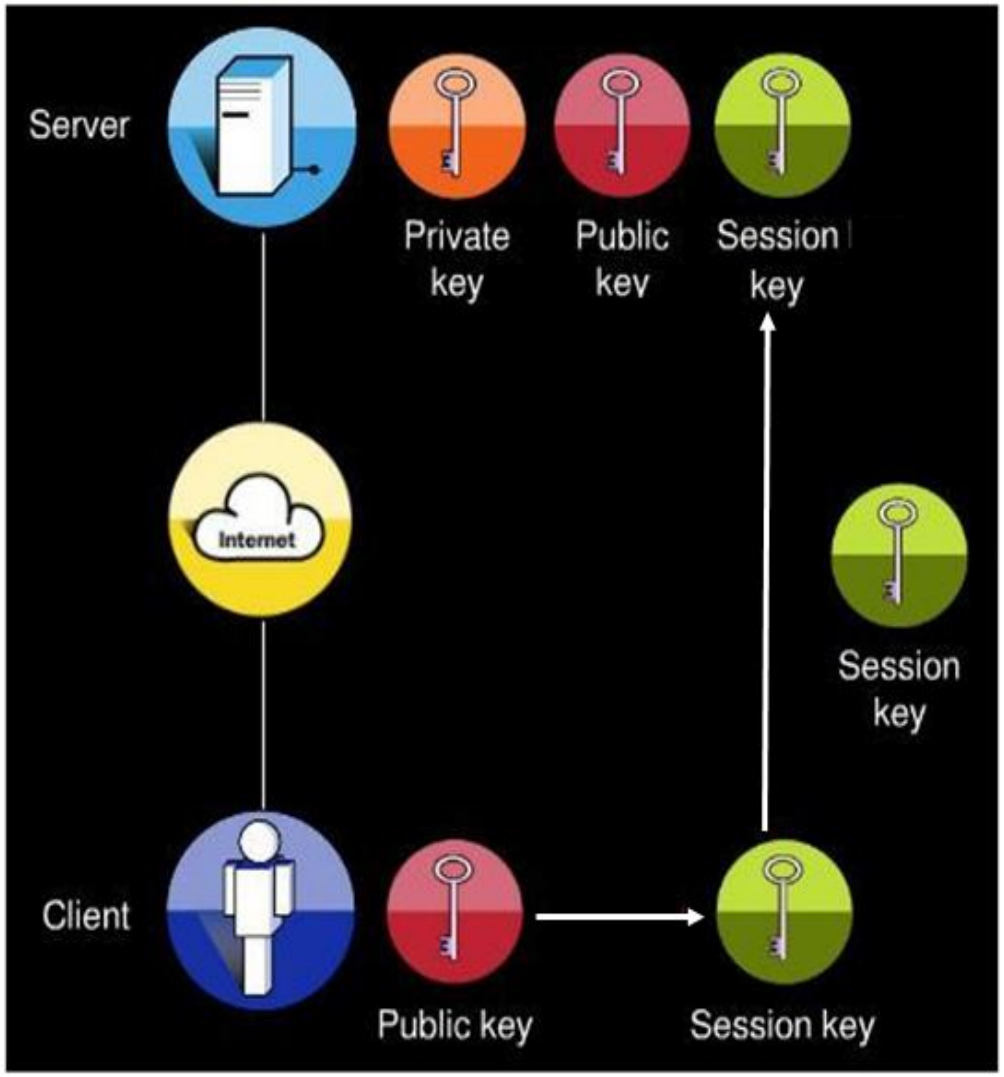



HTTPS

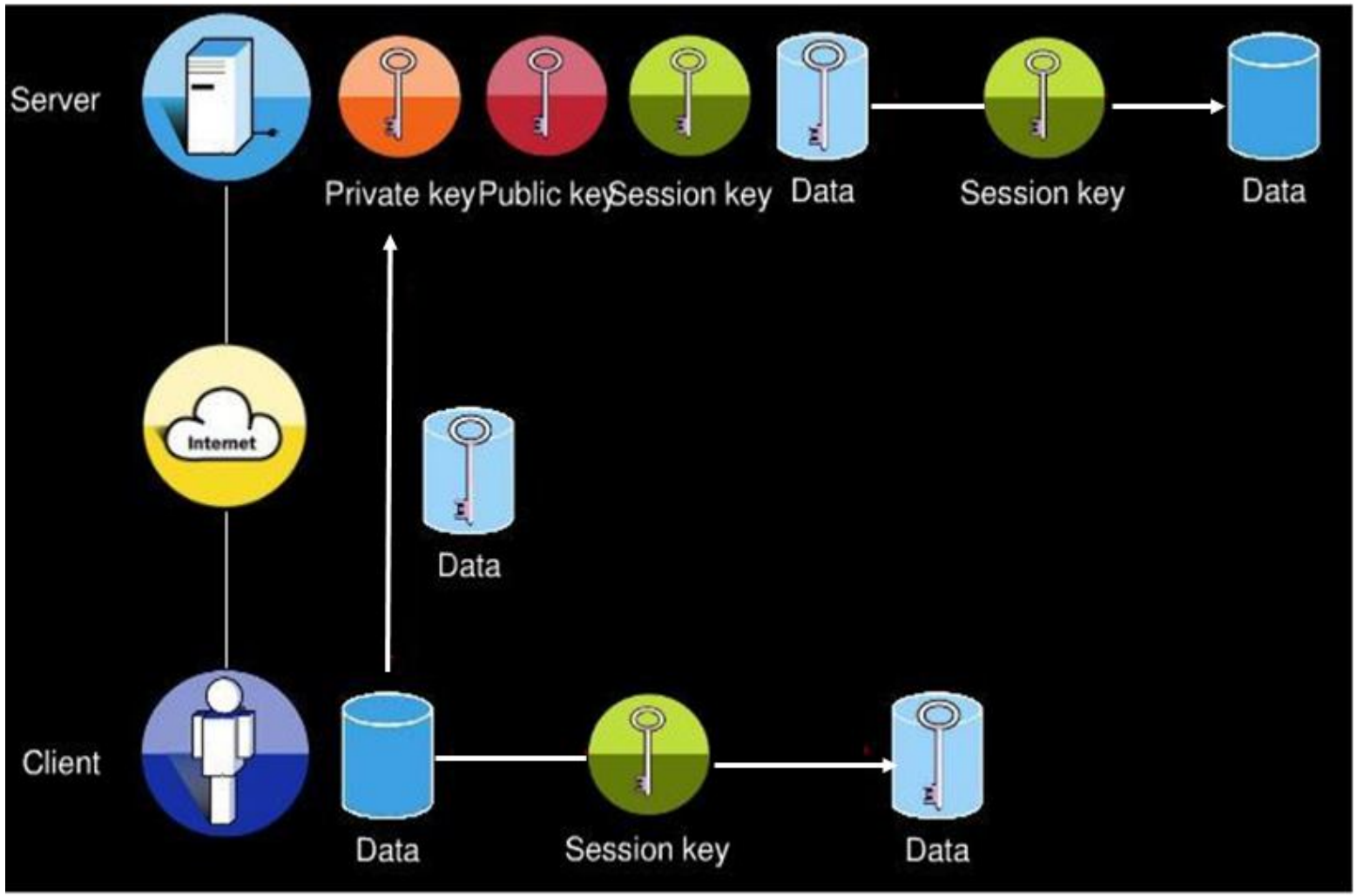
# 2) SSL/TLS Processing(1/3)



# 2) SSL/TLS Processing(2/3)



# 2) SSL/TLS Processing(3/3)



# 3) 공개 키 기반 구조(Public Key Infrastructure, PKI)

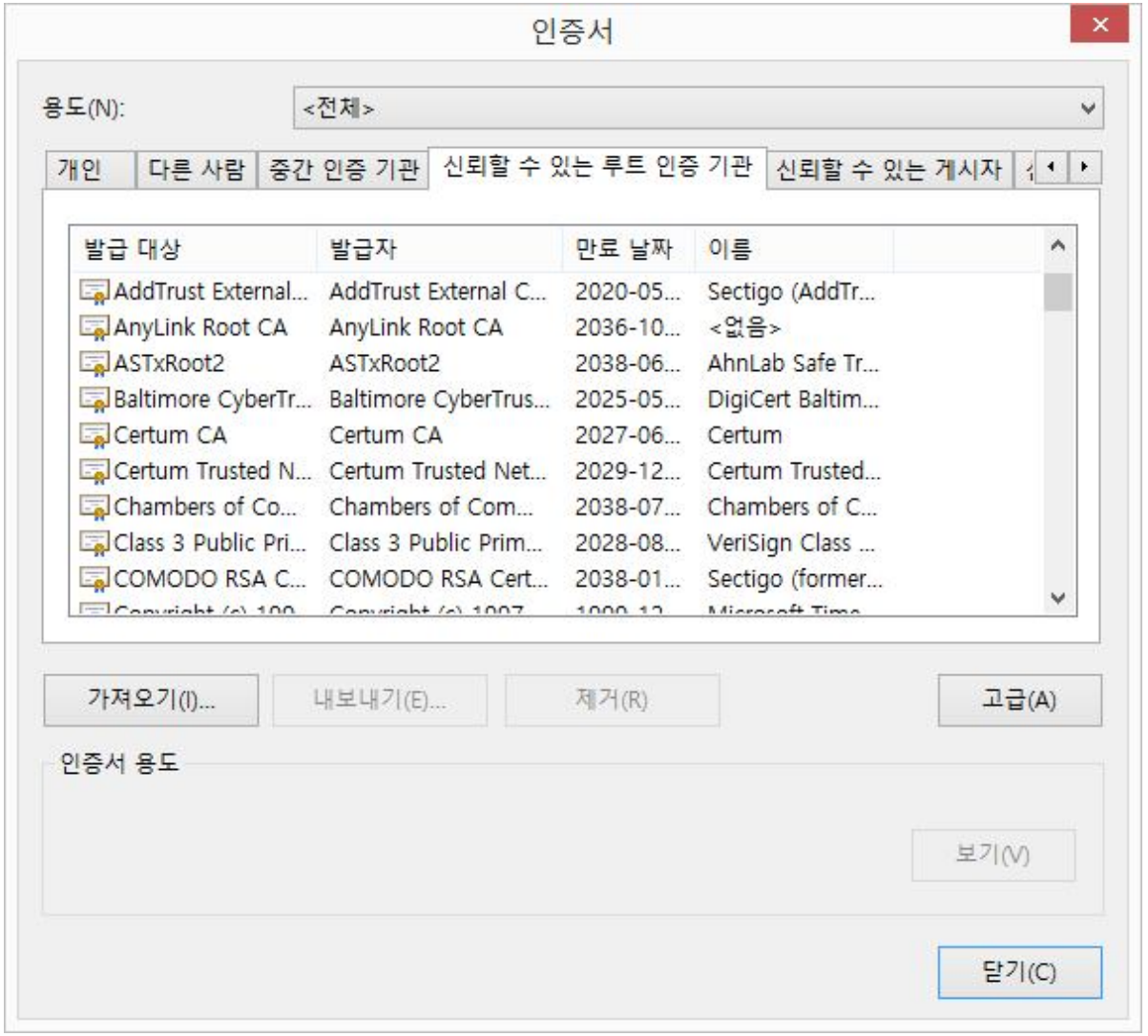
· 메시지의 암호화 및 전자서명을 제공하는 복합적인 보안 시스템환경

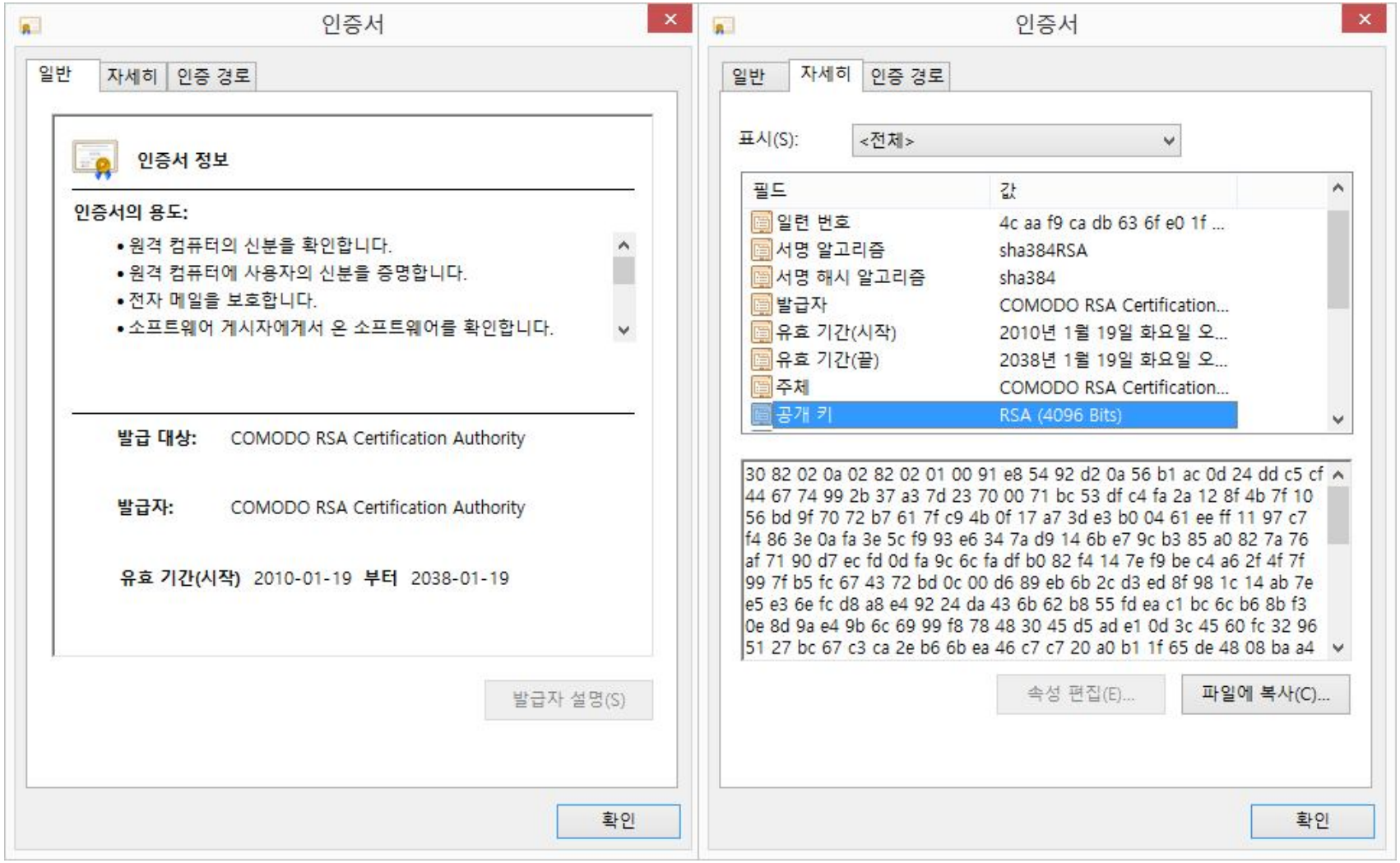
· 공개 키를 효과적으로 운용하기 위해 정해진 많은 규격이나 선택사양

- 디지털 인증서 생성, 관리, 저장, 배분, 취소 (폐지)에 필요한 하드웨어, 소프트웨어, 사람, 정책 및 절차라고 정의

- 공인인증서가 신분증과 같은 효력을 발휘하려면 검증 기관 필요.
- 공개 키 기반 구조는 '인터넷에서 신분증을 검증해주는 관청' 역할.
- 인증 기관(Certification Authority, CA)에서 공인인증서를 이용하여 증명 가능

# ① X.509 인증서


인증서
용도(N):
<전체>
개인
다른 사람
중간 인증 기관
신뢰할 수 있는 루트 인증 기관
신뢰할 수 있는 게시자
발급 대상
발급자
만료 날짜
이름
AddTrust External...
AddTrust External C...
2020-05...
Sectigo (AddTr...
AnyLink Root CA
AnyLink Root CA
2036-10...
<없음>
ASTxRoot2
ASTxRoot2
2038-06...
AhnLab Safe Tr...
Baltimore CyberTr...
Baltimore CyberTrus...
2025-05...
DigiCert Baltim...
Certum CA
Certum CA
2027-06...
Certum
Certum Trusted N...
Certum Trusted Net...
2029-12...
Certum Trusted...
Chambers of Co...
Chambers of Com...
2038-07...
Chambers of C...
Class 3 Public Pri...
Class 3 Public Prim...
2028-08...
VeriSign Class ...
COMODO RSA C...
COMODO RSA Cert...
2038-01...
Sectigo (former...
가져오기(I)...
내보내기(E)...
제거(R)
고급(A)
인증서 용도
보기(V)
닫기(C)


인증서
일반
자세히
인증 경로
인증서 정보
인증서의 용도:
• 원격 컴퓨터의 신분을 확인합니다.
• 원격 컴퓨터에 사용자의 신분을 증명합니다.
• 전자 메일을 보호합니다.
• 소프트웨어 게시자에게서 온 소프트웨어를 확인합니다.
발급 대상: COMODO RSA Certification Authority
발급자: COMODO RSA Certification Authority
유효 기간(시작) 2010-01-19 부터 2038-01-19
발급자 설명(S)
확인
인증서
일반
자세히
인증 경로
표시(S): <전체>
필드
값
일련 번호
4c aa f9 ca db 63 6f e0 1f ...
서명 알고리즘
sha384RSA
서명 해시 알고리즘
sha384
발급자
COMODO RSA Certification...
유효 기간(시작)
2010년 1월 19일 화요일 오...
유효 기간(끝)
2038년 1월 19일 화요일 오...
주체
COMODO RSA Certification...
공개 키
RSA (4096 Bits)
30 82 02 0a 02 82 02 01 00 91 e8 54 92 d2 0a 56 b1 ac 0d 24 dd c5 cf
44 67 74 99 2b 37 a3 7d 23 70 00 71 bc 53 df c4 fa 2a 12 8f 4b 7f 10
56 bd 9f 70 72 b7 61 7f c9 4b 0f 17 a7 3d e3 b0 04 61 ee ff 11 97 c7
f4 86 3e 0a fa 3e 5c f9 93 e6 34 7a d9 14 6b e7 9c b3 85 a0 82 7a 76
af 71 90 d7 ec fd 0d fa 9c 6c fa df b0 82 f4 14 7e f9 be c4 a6 2f 4f 7f
99 7f b5 fc 67 43 72 bd 0c 00 d6 89 eb 6b 2c d3 ed 8f 98 1c 14 ab 7e
e5 e3 6e fc d8 a8 e4 92 24 da 43 6b 62 b8 55 fd ea c1 bc 6c b6 8b f3
0e 8d 9a e4 9b 6c 69 99 f8 78 48 30 45 d5 ad e1 0d 3c 45 60 fc 32 96
51 27 bc 67 c3 ca 2e b6 6b ea 46 c7 c7 20 a0 b1 1f 65 de 48 08 ba a4
속성 편집(E)...
파일에 복사(C)...
확인

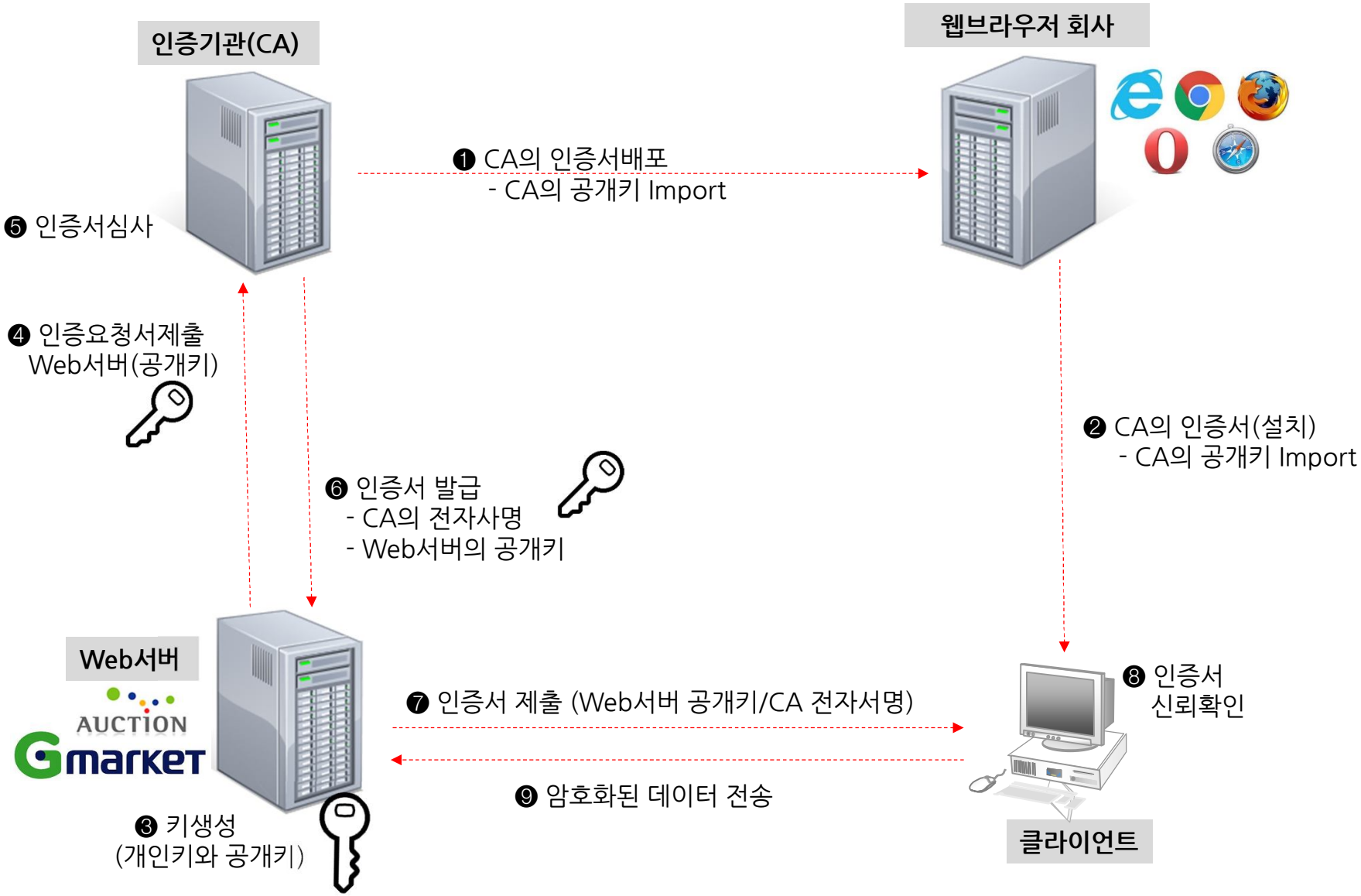
# ② 인증서 발급절차

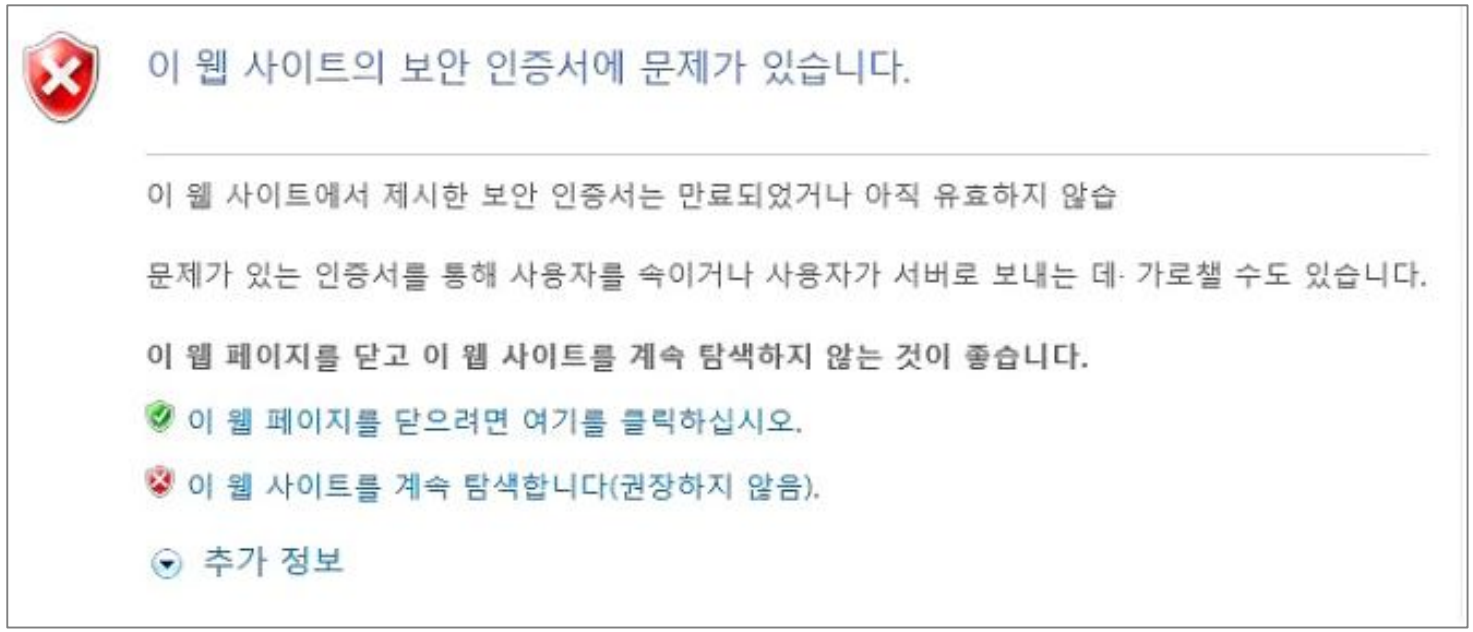
인증기관(CA)

웹브라우저 회사

Web서버


클라이언트


인증기관(CA)
웹브라우저 회사
❶ CA의 인증서배포
- CA의 공개키 Import
❺ 인증서심사
❹ 인증요청서제출
Web서버(공개키)
❷ CA의 인증서(설치)
- CA의 공개키 Import
❻ 인증서 발급
- CA의 전자사명
- Web서버의 공개키
Web서버
AUCTION
Gmarket
❼ 인증서 제출 (Web서버 공개키/CA 전자서명)
❽ 인증서
신뢰확인
❾ 암호화된 데이터 전송
❸ 키생성
(개인키와 공개키)
클라이언트







Web서버

인증서 제출 (Web서버 공개키/CA 전자서명)

클라이언트