



# 안드로이드 키 해시 생성방법

Mac과 Window 환경에서 안드로이드 키 해시를 생성하는 방법을 안내하기 위한 문서입니다.

## Debug

debug.keystore는 android SDK를 설치하면 기본적으로 포함되어 있습니다.

## Mac

### 1. keytool 확인

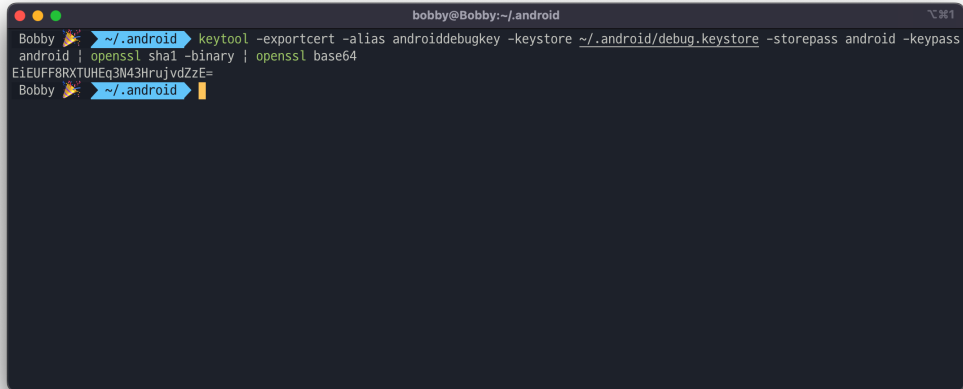
Mac의 경우 JDK 설치시 별도 환경 변수 설정 없이 keytool 사용이 가능합니다.

```
bobby@Bobbyui-MacBookPro:~  
▶ keytool  
키 및 인증서 관리 툴  
  
명령:  
  
-certreq          인증서 요청을 생성합니다.  
-changealias      항목의 별칭을 변경합니다.  
-delete          항목을 삭제합니다.  
-exportcert       인증서를 익스포트합니다.  
-genkeypair       키 쌍을 생성합니다.  
-genseckey        보안 키를 생성합니다.  
-gencert          인증서 요청에서 인증서를 생성합니다.  
-importcert       인증서 또는 인증서 체인을 임포트합니다.  
-importpass       비밀번호를 임포트합니다.  
-importkeystore   다른 키 저장소에서 하나 또는 모든 항목을 임포트합니다.  
-keypasswd        항목의 키 비밀번호를 변경합니다.  
-list            키 저장소의 항목을 나열합니다.  
-printcert        인증서의 콘텐츠를 인쇄합니다.  
-printcertreq     인증서 요청의 콘텐츠를 인쇄합니다.  
-printcrl         CRL 파일의 콘텐츠를 인쇄합니다.  
-storepasswd      키 저장소의 저장소 비밀번호를 변경합니다.  
  
사용 가능한 모든 명령에 "keytool -help" 사용  
command_name 사용법에 "keytool -command_name -help"를 사용합니다.  
-conf <url> 옵션을 사용하여 사전 구성된 옵션 파일을 지정합니다.  
Bobby 🐼 🚀  
▶
```

### 2. 터미널에서 명령어 실행

```
keytool -exportcert -alias androiddebugkey \  
-keystore ~/.android/debug.keystore \
```

```
-storepass android \  
-keypass android | openssl sha1 -binary | openssl base64
```

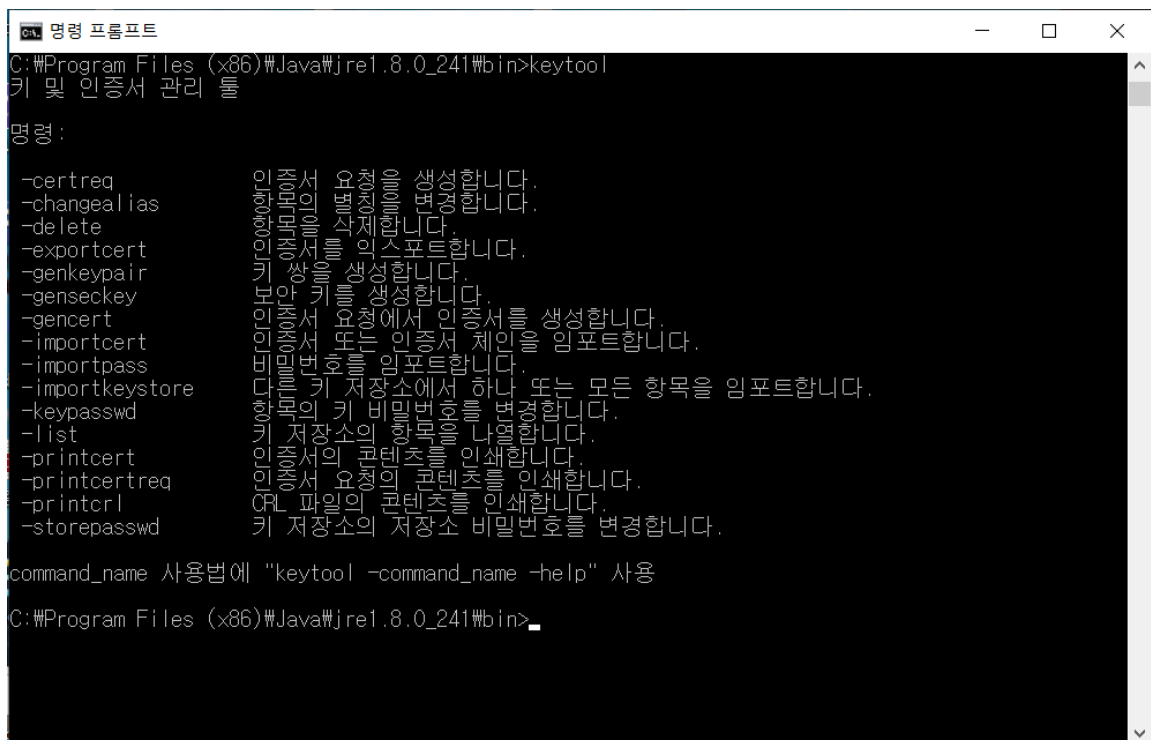


```
bobby@Bobby:~/android  
Bobby ~$ keytool -exportcert -alias androiddebugkey -keystore ~/android/debug.keystore -storepass android -keypass  
android | openssl sha1 -binary | openssl base64  
EiEUFF8RXTUHEq3N43HrujvdZzE=  
Bobby ~$
```

## Window

### 1. keytool 확인

윈도우의 경우 jdk/bin 디렉터리를 환경변수 PATH에 설정하거나, 터미널 상에서 jdk/bin 경로로 이동 후 keytool 명령어를 사용할 수 있습니다.




```
명령 프롬프트  
C:\Program Files (x86)\Java\jre1.8.0_241\bin>keytool  
키 및 인증서 관리 툴  
  
명령 :  
-certreq          인증서 요청을 생성합니다.  
-changealias      항목의 별칭을 변경합니다.  
-delete           항목을 삭제합니다.  
-exportcert       인증서를 익스포트합니다.  
-genkeypair       키 쌍을 생성합니다.  
-genseckey        보안 키를 생성합니다.  
-gencert          인증서 요청에서 인증서를 생성합니다.  
-importcert       인증서 또는 인증서 체인을 임포트합니다.  
-importpass       비밀번호를 임포트합니다.  
-importkeystore   다른 키 저장소에서 하나 또는 모든 항목을 임포트합니다.  
-keypasswd        항목의 키 비밀번호를 변경합니다.  
-list             키 저장소의 항목을 나열합니다.  
-printcert        인증서의 콘텐츠를 인쇄합니다.  
-printcertreq     인증서 요청의 콘텐츠를 인쇄합니다.  
-printcrl         CRL 파일의 콘텐츠를 인쇄합니다.  
-storepasswd      키 저장소의 저장소 비밀번호를 변경합니다.  
  
command_name 사용법에 "keytool -command_name -help" 사용  
C:\Program Files (x86)\Java\jre1.8.0_241\bin>
```

### 2. OpenSSL 설치

1) 아래 주소로 이동하여 OpenSSL을 설치

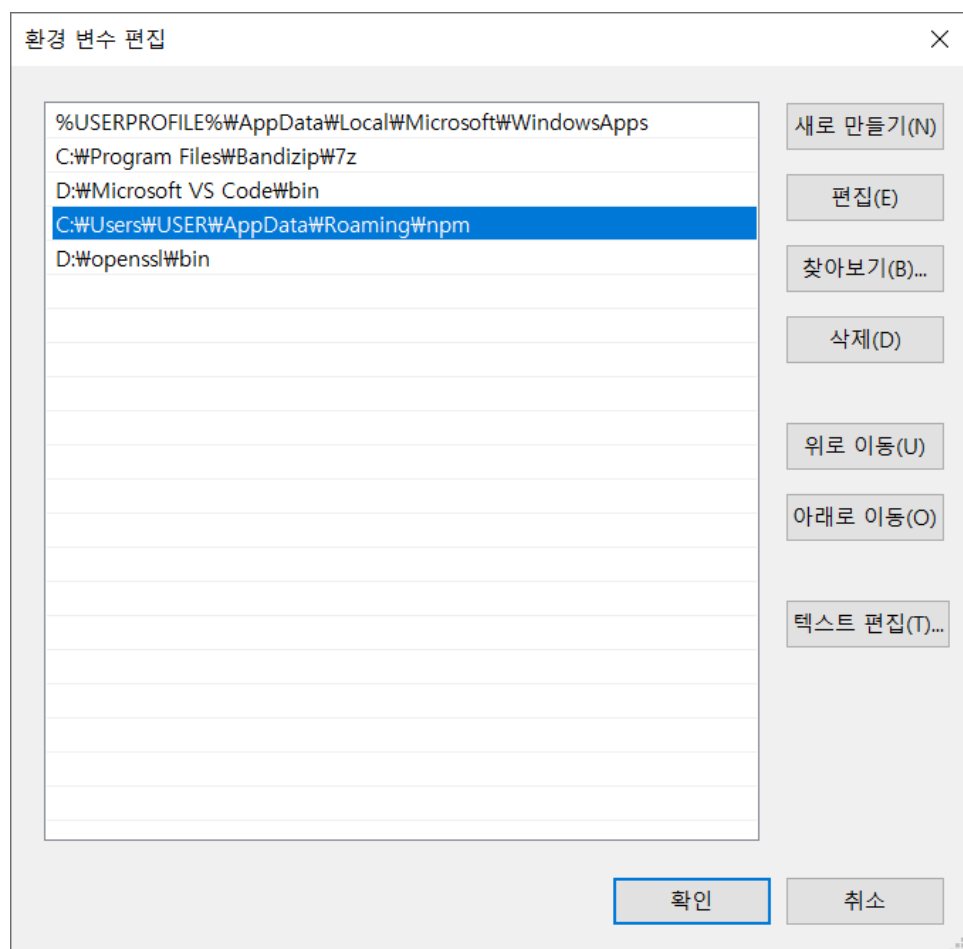
[code.google.com/archive/p/openssl-for-windows/downloads](https://code.google.com/archive/p/openssl-for-windows/downloads)

가 붙은 버전을 다운받길 추천드립니다.

<a href="#">openssl-0.9.8e_X64.zip</a>	openssl-0.9.8e X64	Jul 23, 2009	1.25MB
<a href="#">openssl-0.9.8e_WIN32.zip</a>	openssl-0.9.8e WIN32	Jul 23, 2009	1.08MB

2) 다운 받은 파일의 압축을 풀고 /bin 디렉토리 경로를 환경 변수에 추가

ex) openssl\bin



3. 터미널에서 명령어 실행

```
keytool -exportcert -alias androiddebugkey \  
-keystore %HOMEPATH%\android/debug.keystore \  
-storepass android \  
-keypass android | openssl sha1 -binary | openssl base64
```

```
명령 프롬프트
C:\Program Files (x86)\Java\jre1.8.0_241\bin>keytool -exportcert -alias androiddebugkey -keystore %HOMEPATH%\androiddebug.keystore -storepass android -keypass android | openssl sha1 -binary | openssl base64
Z6eLfsvMMvD8GBf/IqmhrTLNQd4=
C:\Program Files (x86)\Java\jre1.8.0_241\bin>
```

추가적으로 릴리즈 키 해시 생성 방법도 안내합니다.

## Release

릴리즈 키 해시를 생성하기 위해선, release.keystore 가 필요합니다.

release keystore는 jdk의 keytool 을 이용하여 생성가능하며 release.keystore는 생성된 후 외부로 노출되지 않도록 각별한 주의를 필요로 합니다!

## Mac

### 1. 명령어를 통해 release keystore 생성

```
keytool -genkey -v \
  -keystore ~/<키스토어 파일 이름>.jks \
  -keyalg RSA -keysize 2048 \
  -validity 10000 -alias <alias>
```

```
bobby@Bobby:~  
x Bobby keytool -genkey -v -keystore ~/upload.keystore.jks -keyalg RSA -keysize 2048 -validity 10000 -alias upload  
키 저장소 비밀번호 입력:  
새 비밀번호 다시 입력:  
이름과 성을 입력하십시오.  
[Unknown]: 신현욱  
조직 단위 이름을 입력하십시오.  
[Unknown]:  
조직 이름을 입력하십시오.  
[Unknown]:  
구/군/시 이름을 입력하십시오?  
[Unknown]:  
시/도 이름을 입력하십시오.  
[Unknown]:  
이 조직의 두 자리 국가 코드를 입력하십시오.  
[Unknown]: KR  
CN=신현욱, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=KR이(가) 맞습니까?  
[아니오]: Y  
  
다음에 대해 유효 기간이 10,000일인 2,048비트 RSA 키 쌍 및 자체 서명된 인증서(SHA256withRSA)를 생성하는 중  
: CN=신현욱, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=KR  
[/Users/bobby/upload.keystore.jks를(를) 저장하는 중]  
Bobby
```

alias와 password를 분실하지 않도록 주의해야합니다!

## 2. release keystore로 key hash 확인

```
keytool -exportcert -alias <RELEASE_KEY_ALIAS> \  
-keystore <RELEASE_KEY_PATH> \  
| openssl sha1 -binary | openssl base64
```

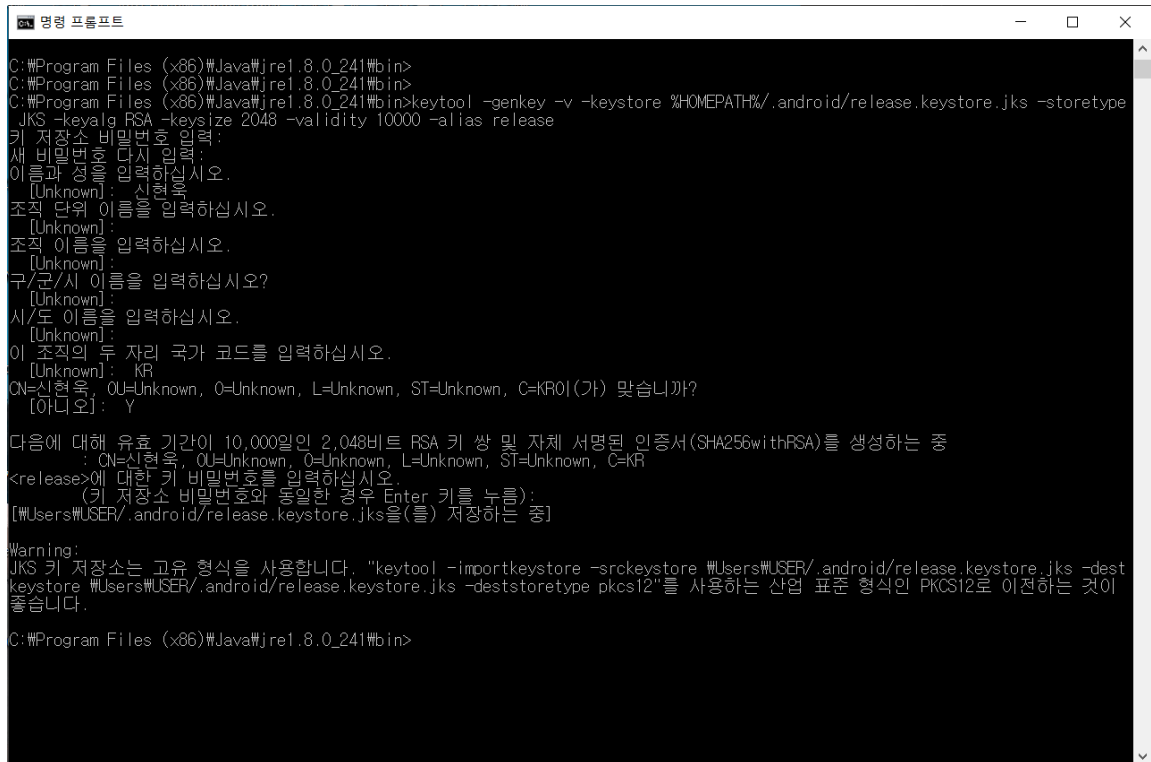
```
bobby@Bobby:~  
Bobby keytool -exportcert -alias upload -keystore ~/upload.keystore.jks | openssl sha1 -binary | openssl base64  
키 저장소 비밀번호 입력:   
4nRYwIpPzQfIXdgr+AhR43J5D1Q=  
Bobby
```

## Window

### 1. 명령어를 통해 release keystore 생성

```
keytool -genkey -v -keystore %HOMEPATH%\android\<키스토어 파일 이름>.jks \
    -storetype JKS -keyalg RSA -keysize 2048 -validity 10000 \
    -alias <alias>
```

alias와 password를 분실하지 않도록 주의해야합니다!



## 2. release keystore로 key hash 확인

```
keytool -exportcert -alias <RELEASE_KEY_ALIAS> \
    -keystore <RELEASE_KEY_PATH> \
    | openssl sha1 -binary | openssl base64
```

```
명령 프롬프트
C:\Program Files (x86)\Java\jre1.8.0_241\bin>keytool -exportcert -alias release -keystore %HOMEPATH%\android/release.keystore.jks | openssl sha1 -binary | openssl base64
키 저장소 비밀번호 입력:

Warning:
JKS 키 저장소는 고유 형식을 사용합니다. "keytool -importkeystore -srckeystore %USER%\android/release.keystore.jks -destkeystore %USER%\android/release.keystore.jks -deststoretype pkcs12"를 사용하는 산업 표준 형식인 PKCS12로 이전하는 것이 좋습니다.
m2vrSv8CEDHrdqUoIRu0X1tM9A=
C:\Program Files (x86)\Java\jre1.8.0_241\bin>
```