SSH Key File Formats

Beshir Said

Private Key

- I expect to find (p, q, n, e, d). I took the PEM file and uploaded it to a ASN.1 decoder to get the DER. The output showed 9 INTEGER values corresponding to the version, modulus, public exponent, private exponent, prime1, prime2, exponent1, exponent2, and coefficient (Figure 1).

ASN.1 name	Meaning	Value (decimal)
version	Tells us the format of the data	0
modulus	This is n where n = p x q	(3072 bit) 533506519430170549384581495278725897151476037577546980387977095222584456643698582550150358703919158080560775989607298393292955564723921068005109047687770896884905650961373845008928198279307506079014612068982086247432257609007718408548331372300422506016111804528615642110338767715673822046589759710906679747956088069746913799863420659084213971935676595304358023712682514256774416655919860319007317394016128080338071582348413810495911117833987028541133304227738334123327023459232916175257074154169198915626599378449755329902423562518441014971973111160057895054700871573131740292559014592839138900390113555087079627683786700899122718142699397029960262048228258868790549517649965467869270073982543218989893527860416826240121128988593646899471216268725231522469130459707273095437778893458400362302368364461544595891088085626628070469661919663611682160906596063973729513975114328139205403181703561685657975520291419647125462156171
publicExponent	This is e, public part of key	65537
privateExponent	This is d, secret used to descrypt	(3072 bit) 393196185284597520573487813043894892325151351008332647202644028203092174502148179436880120783038577512690023056807942379029297141030717175118097748025802555662264489480527306525721972398627215940956784374382467122956886564424398535961257817927013861835058307071374401171421835290519268179384652696896463654211010724580705303068542674434695195972734718247643268738976122207553917617492182633754557539227810281343503601004176807921677875131596006310409077734775611550382903008291662005396657491732428929653132099447168773239600116292809334095108435023354763663488412068339319890270213300189687171913185953672281161789399044474127352289341523096780950566384384521241129919214370701496933795679678976982904744573355892734350195126251252635731292071942749574053366330738244276046930364495545544567286609547262276228227492793593774913561670308180160797466929634915091701753717512944184198332885383671520194777891216475403071383572 9

prime1	This is p (first prime)	(1536 bit) 232491389517685016006360623390512529503742646502874318487803875699755357020970661152255688165234969193273796488705107942297769169723067694994007419978275345576744531239113114492998628598617961678532110409693032129758668468494030172904048300748881468863448612322916870767802570162242650113836826915282820939739169140621034011925756509152296597267055270143894542937988134421371340447688225011206237272418292445831611359845477089905679299497542735030637848129003827
prime2	This is q (second prime)	(1536 bit) 229473668051516417324903971476775068725300293771178652098681151270115274521555790716981312212455411269576477463805652787185972196290929290720127706470381260627318068750991105878769044481854954818502171953006217549738641091632591878376585686519375543786022267822696862558524078096146964706331120870500428448113327753501965419996742591602357101160600702657194393266638142787571981865622718145701702293877145886678521275895367109273725096645604040050110871811299847
exponent1	This is d mod (p-1)	(1535 bit) 653907530552129011099874173513788769144527244608004991483541944363274256674481925785362329668702593518106701661092398904492848178205488017734186607940483916721261568935192645291887593414151530833022932264502748256380599642911707612057360319774193526642957149541225060479256721516184562849742012989518628924456735091485347197508696915226578622418231377612072693439979481541558782343900957499299091683209276968033539713383763052904945092507105989514621472672870707
exponent2	This is d mod (q-1)	(1536 bit) 163783210961407021023081742676619781416522674846890346377892023312188719059772245702084591300331329135384733964737067836083602170586149938244422137080710954836251735415053480760262154579600033059039284606768982896784635788977312322854924272893637341290186880688381653768672998108326142806404695207875368739627218502491843302039575071559452763536449005409050230243072856051260616502836081358367682782220690235977760825202277978903333462329596633617406749467405612 9
coefficient	This is the value that helps combine results (CRT)	(1536 bit) 224742649770660865757565783761247504395531622887536240696894763609774888767056605435924267982074049835890473730939547470899626218160361830402907094488358499800193334300396650194681744090273225898661289438192998012540378226952528573766205845794505026597933891541013086426769109511323315159132534894112778967816004493827647801786499029784361529529153120606803002462454471451871476238515267327198512630216158021168736650240879592837614774152252504939252375892572348 6

Table 1: Field results for RSA private key

Public Key

- I expect to get the e and n. I took the OpenSSL format file and converted the pub file to a PEM form so that I can use the ASN.1 decoder. I got 2 elements: Version and modulus.

ASN.1 name	Meaning	Value
modulus	This is n	(3072 bit) 533506519430170549384581495278725897151476037577546980387977 095222584456643698582550150358703919158080560775989607298393 292955564723921068005109047687770896884905650961373845008928 198279307506079014612068982086247432257609007718408548331372 300422506016111804528615642110338767715673822046589759710906 679747956088069746913799863420659084213971935676595304358023 712682514256774416655919860319007317394016128080338071582348 413810495911117833987028541133304227738334123327023459232916 175257074154169198915626599378449755329902423562518441014971 973111160057895054700871573131740292559014592839138900390113 555087079627683786700899122718142699397029960262048228258868 790549517649965467869270073982543218989893527860416826240121 128988593646899471216268725231522469130459707273095437778893 458400362302368364461544593589108808562662807046966191966361 168216090659606397372951397511432813920540318170356168565797 552029141964712546215617 1
publicExponent	This is e	65537

Sanity Check

- To confirm that the values we got using the two files are exactly the same if we would've used RSA, we can verify each of the properties. To check we could see if the modulus, n, are the same for the public and private keys and see if p x q = n. We can also check if the $\lambda(n)=lcm(p-1,q-1)$ and can verify it using $e \cdot d = 1 (mod \lambda(n))e = 1 \bmod \{\lambda(n)\} e \cdot d = 1 (mod \lambda(n))$. After calculations, we can verify that by multiplying p with q this gives us modulus , n, exactly and that the public exponent ,e, and private exponent ,d, are multiplicative inverse of mod lambda n . We also know from the experiment that the private key and public key share the n and e Therefore, we can conclude that the private and public key form a valid RSA key pair.

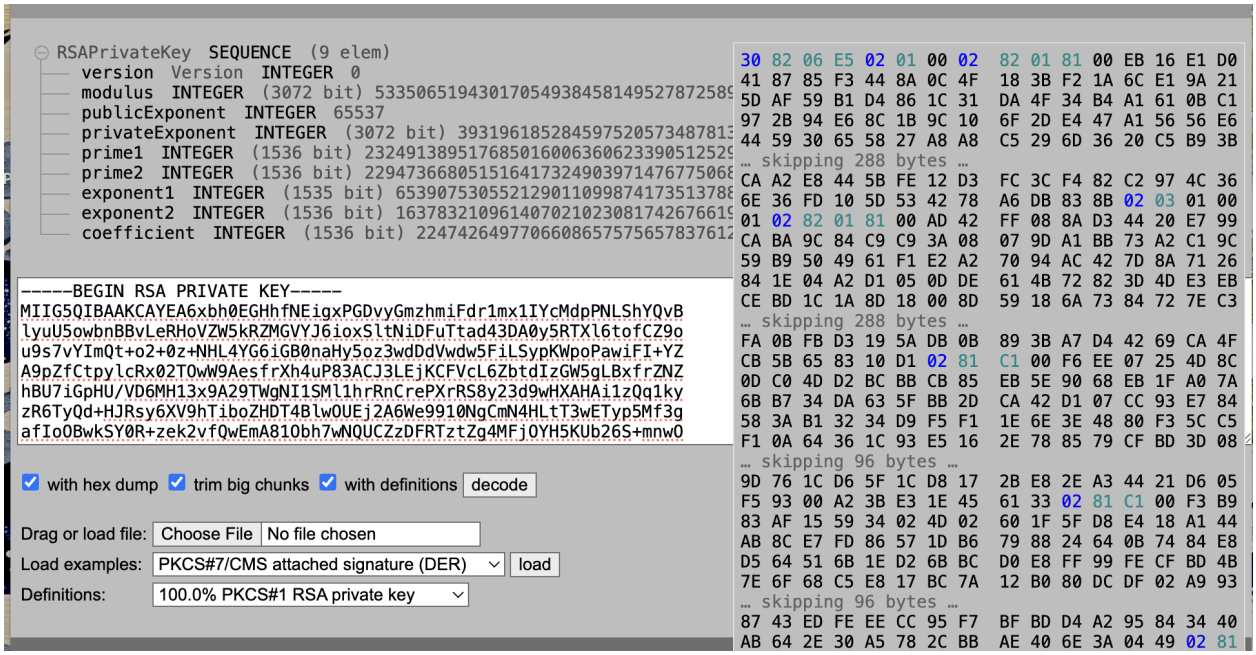

Figure 1: Results of the RSA Private Key insertion io the ASN.1 decoder