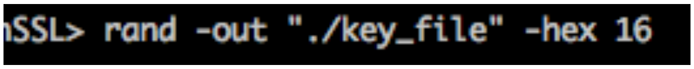
Tyler Jackson

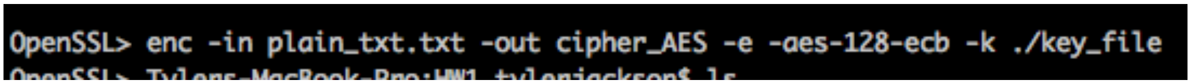
CSE 7349 HW 1

generate the 128-bit key



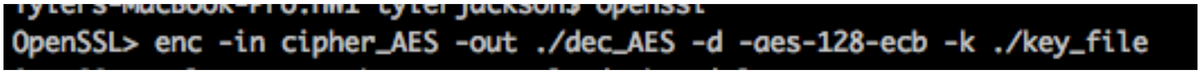
```
nSSL> rand -out "./key_file" -hex 16
```

generate the cipher text using the key and AES 128 bit ECB cipher



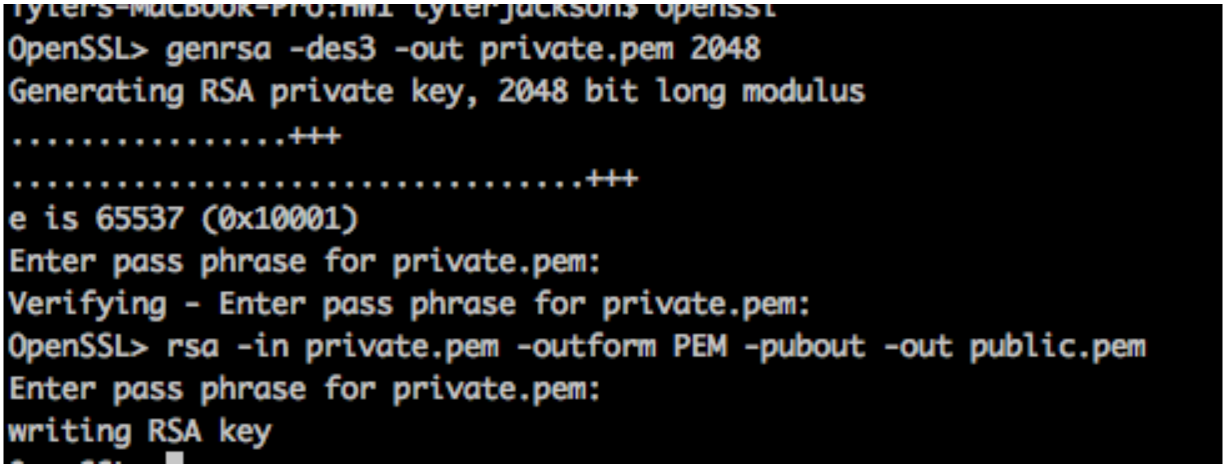
```
OpenSSL> enc -in plain_txt.txt -out cipher_AES -e -aes-128-ecb -k ./key_file
OpenSSL> Tylers-MacBook-Pro:HW1 tylerjackson$ ls
```

decrypt the cipher text using the same symmetric key and store the plaintext in dec_AES
-verified that the contents were equivalent to the original plain_txt



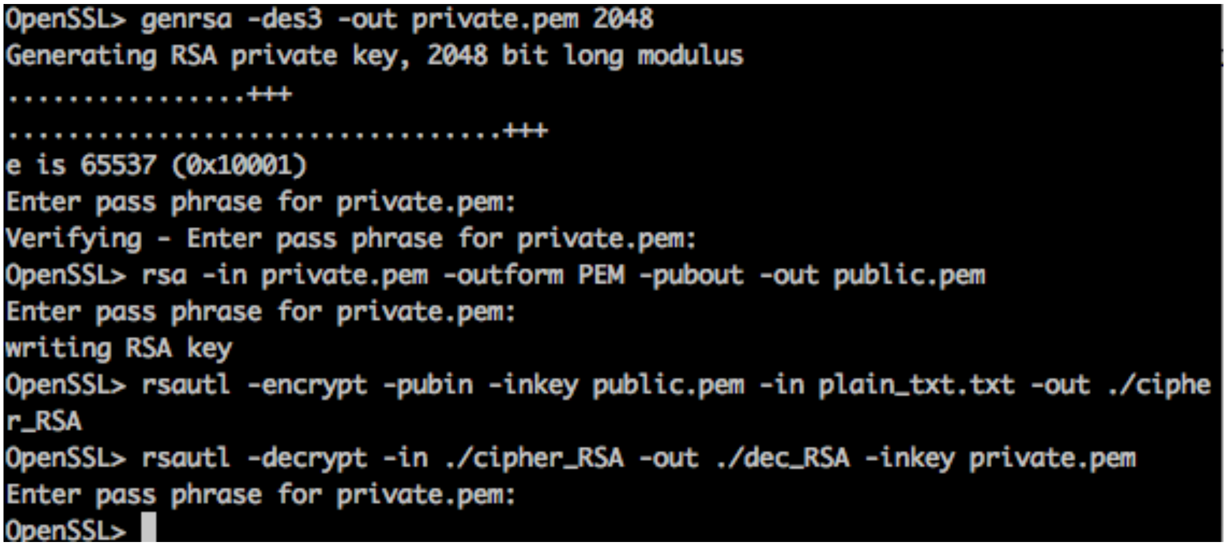
```
Tylers-MacBook-Pro:HW1 tylerjackson$ openssl
OpenSSL> enc -in cipher_AES -out ./dec_AES -d -aes-128-ecb -k ./key_file
```

Use RSA to generate a 2048 bit private key
-then extract the public key from the private key file



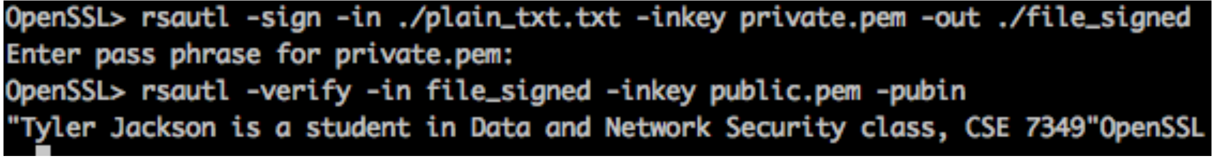
```
Tylers-MacBook-Pro:HW1 tylerjackson$ openssl
OpenSSL> genrsa -des3 -out private.pem 2048
Generating RSA private key, 2048 bit long modulus
................+++
.................................+++
e is 65537 (0x10001)
Enter pass phrase for private.pem:
Verifying - Enter pass phrase for private.pem:
OpenSSL> rsa -in private.pem -outform PEM -pubout -out public.pem
Enter pass phrase for private.pem:
writing RSA key
OpenSSL> 
```

encrypt the plain_txt using the public key and decrypt using the private key
-verified contents were the same



```
OpenSSL> genrsa -des3 -out private.pem 2048
Generating RSA private key, 2048 bit long modulus
................+++
..................................+++
e is 65537 (0x10001)
Enter pass phrase for private.pem:
Verifying - Enter pass phrase for private.pem:
OpenSSL> rsa -in private.pem -outform PEM -pubout -out public.pem
Enter pass phrase for private.pem:
writing RSA key
OpenSSL> rsautl -encrypt -pubin -inkey public.pem -in plain_txt.txt -out ./ciphe
r_RSA
OpenSSL> rsautl -decrypt -in ./cipher_RSA -out ./dec_RSA -inkey private.pem
Enter pass phrase for private.pem:
OpenSSL>
```

Sign the message using private key and verify using public key - as you can see the private and public keys were a pair.



```
OpenSSL> rsautl -sign -in ./plain_txt.txt -inkey private.pem -out ./file_signed
Enter pass phrase for private.pem:
OpenSSL> rsautl -verify -in file_signed -inkey public.pem -pubin
"Tyler Jackson is a student in Data and Network Security class, CSE 7349"OpenSSL
```