How to encrypt a big file using OpenSSL and someone's public key

The situation

You have a public key for someone, you have a file you want to send them, you want to send it securely.

Can you call them, securely chat with them, or send them an encrypted e-mail?

If you are set up to chat over OTR with them or to send them an encrypted e-mail, just use that to send your file. It'll be faster. If you can call them, then call them and agree on a symmetric key. Then just use that key to encrypt the file like this. If you can't (or don't want to) do either of those, then you can follow this how-to.

Step 0) Get their public key

The other person needs to send you their public key in .pem format. If they only have it in rsa format (*e.g.*, they use it for ssh), then have them do:

```
openssl rsa -in id_rsa -outform pem > id_rsa.pem

openssl rsa -in id_rsa -pubout -outform pem > id_rsa.pub.pem
```

Have them send you *id_rsa.pub.pem*

Step 1) Generate a 256 bit (32 byte) random key

```
openssl rand -base64 32 > key.bin
```

Step 2) Encrypt the key

```
openssl rsautl -encrypt -inkey id_rsa.pub.pem -pubin -in key.bin -out key.bin.enc
```

Step 3) Actually Encrypt our large file

```
openssl enc -aes-256-cbc -salt -in SECRET_FILE -out SECRET_FILE.enc -pass
file:./key.bin
```

Step 4) Send/Decrypt the files

Send the *.enc* files to the other person and have them do:

```
openssl rsautl -decrypt -inkey id_rsa.pem -in key.bin.enc -out key.bin

openssl enc -d -aes-256-cbc -in SECRET_FILE.enc -out SECRET_FILE -pass
file:./key.bin
```

Notes

You should **always** verify the hash of the file with the recipient or sign it with your private key, so the other person knows it actually came from you.

If there is a man-in-the-middle, then he/she could substitute the other person's public key for his/her own and then you're screwed. **Always** verify the other person's public key (take a hash and read it to each other over the phone).

■