Segurança Informática e de Redes de Computadores

Nome: Mateus Fernando Jó Matazeus

LAB1

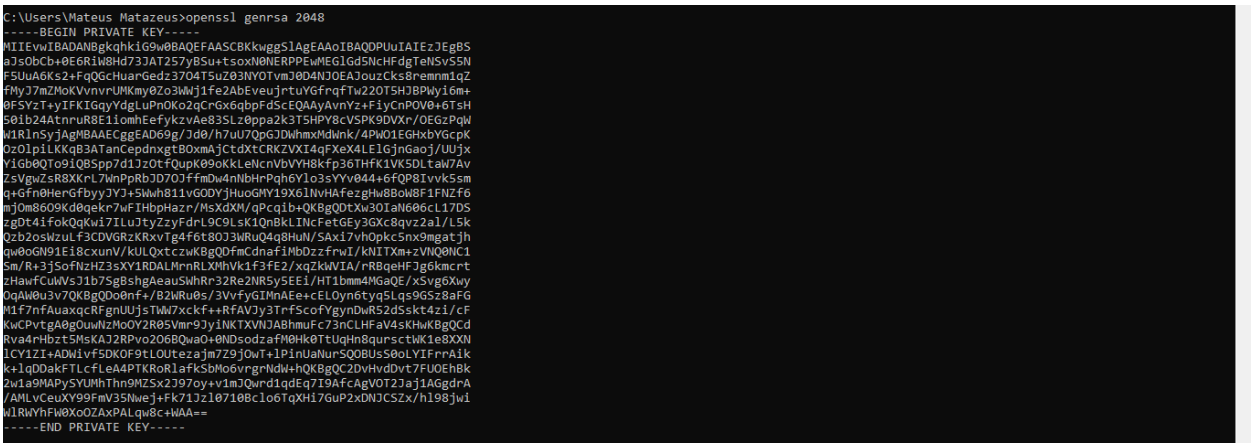
1. Geração de Chaves RSA de Vários Tamanhos:
   - 512 bits:

Executando o comando openssl genrsa 512 gerou com sucesso uma chave RSA de 512 bits exibida na imagem acima.

   - 512 bits:

Executando o comando openssl genrsa 1024 gerou com sucesso uma chave RSA de 1024 bits exibida na imagem acima.

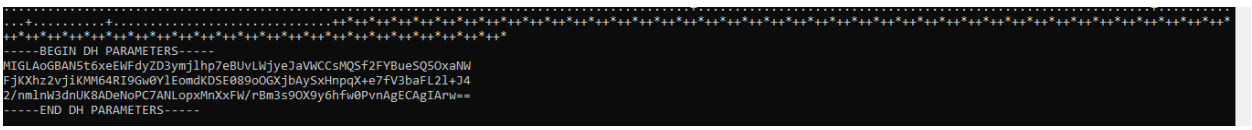
   - 2048 bits:



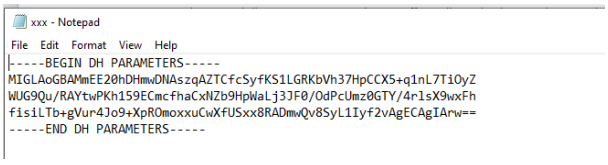
Executando o comando openssl genrsa 2048 gerou com sucesso uma chave RSA de 2048 bits exibida na imagem acima.

Para fazer a compracao com a algoritmo Diffie–Hellman (DH) fez-se a execução do comando openssl gendh 1024 e obteve-se o seguinte resultado:

Segundo pesquisas feitas, conclui-se que o erro acima ocorre pelo facto do comando openssl gendh não está disponível nas versões mais recentes do OpenSSL. Recomenda-se fazer o uso do comando openssl dhparam para gerar chaves Diffie-Hellman (DH).
Fazendo o uso do comando openssl dhparam 1024 -out xxx obteve-se a chave:

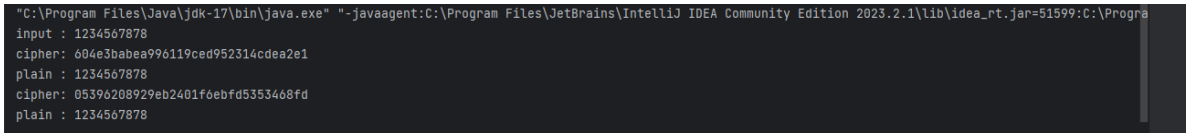


O comando openssl dhparam 1024 -out xxx gerou a chave Diffie-Hellman de 1024 bits acima e o salvou em um arquivo de nome"xxx". Pode-se conferir a chave gerado na imagem abaixo.

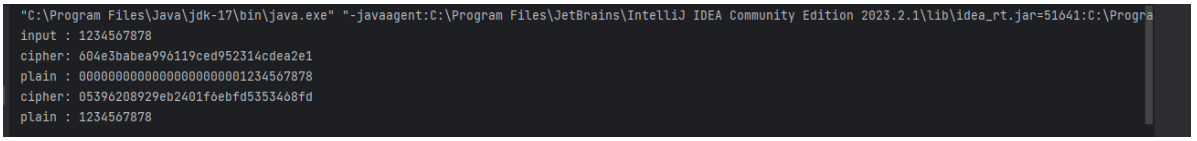


2.1. Olhando o codigo BaseRSAExample fornecido na turma, nota-se que o mesmo mostra a habilidade de cifrar e decifrar informações usando um conjunto de chaves geradas usando o algoritmo assimetrico RSA. Ele demonstra como é possível codificar com a chave pública e decodificar com a chave privada, e vice-versa. O codigo BaseRSAExample ajuda a compreender o funcionamento básico da criptografia RSA.

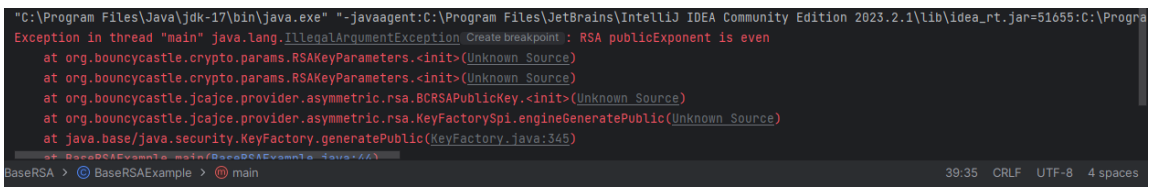
a) As chaves usadas possuem um tamanho de 128 bits, uma vez que o valor do módulo usado que está em formato hexadecimal é d46f473a2d746537de2056ae3092c451.
b) No código fornecido, as chaves estão sendo inicializadas manualmente nas variáveis pubKeySpec e privKeySpec. O valor do módulo (N) é d46f473a2d746537de2056ae3092c451 e o expoente público é 11 (em hexadecimal) e o expoente privado é 57791d5430d593164082036ad8b29fb1.
c) Sim, o tamanho do ciphertext produzido faz sentido. O aumento do tamanho é devido ao uso das técnicas criptograficas, pode-se notar na figura abaixo o resultado



d) Após fazer as alteraçoes referidas, tes-se o resultado abaixo:



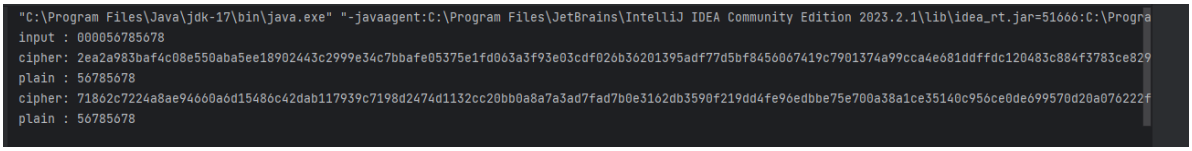
e) Se o bloco plaintext for maior que o valor N subjacente à operação mod e às chaves usadas, a operação de cifragem falhará. A cifragem RSA requer que o plaintext seja menor que o valor N para funcionar corretamente. Apos fazer os testes, obteve-se:



f) No código fornecido na turma, não está sendo usado padding. Isso pode levar a problemas de segurança. O padding ajuda a garantir a segurança e a integridade da mensagem criptografada, prevenindo ataques como o padding oracle, onde um atacante pode inferir informações sobre a mensagem original ao analisar as respostas da decifragem.

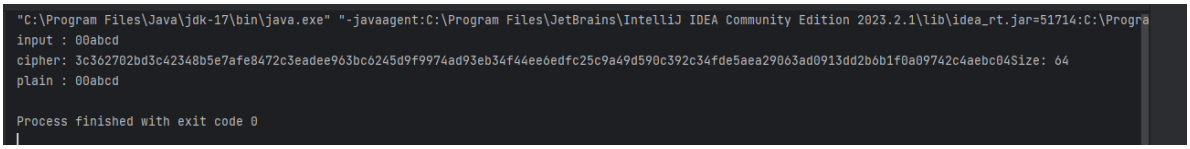
2.2. a) O código RandomKeyRSA gera um par de chaves RSA aleatórias de 8192 bits usando KeyPairGenerator. O código anterior utilizava chaves com valores fixos para fins de demonstração, enquanto este código gera chaves aleatórias.

b) O desempenho da geração de chaves do código RandomKeyRSA e do tempo de computação (cifra/decifra) aumenta significativamente devido ao similar aumento do tamanho da chave. Ou seja, quanto maior for a chave, maior sera o tempo de geracao de chaves. Na figura abaixo verifica-se o grande tamanho do ciphertext:



2.3. a) O codigo PKCS1PaddedRSA utiliza a forma V1 do padding PKCS1. A escolha de usar o padding PKCS1 é importante para garantir que as mensagens tenham uma estrutura padronizada antes da criptografia, o que ajuda a evitar vulnerabilidades e ataques conhecidos em RSA.

2.4. a) O tamanho do ciphertext obtido é o esperado. Isso ocorre porque o tamanho do ciphertext em RSA depende principalmente do tamanho da chave RSA e não do método de preenchimento. Pode-se verificar o resultado abaixo:



b)) O programa OAEPPaddingRSA gera um novo par de chaves RSA em cada execução. Isso faz sentido porque a segurança do RSA depende da escolha adequada das chaves. As operações de padding OAEP são realizadas com base nas chaves geradas, garantindo que cada corrida do programa use chaves diferentes para criptografar e descriptografar. Portanto, a inicialização do cálculo de padding faz sentido mesmo ao gerar novos pares de chaves a cada execução.

Exercicio Opcional

a) No protocolo descrito, a distribuição de KS1 e KS2 é feita com base em envelopes de chave pública. Ou seja, para enviar essas chaves simétricas para o destinatário, o remetente as criptografa usando a chave pública do destinatário. O destinatário, que possui a chave privada correspondente, pode então descriptografar as chaves simétricas e usá-las para comunicação segura.
b) A e B podem ter razoável confiança de que estão se comunicando com os interlocutores corretos, desde que o sistema de chave pública seja seguro e que a autenticidade das chaves públicas seja verificada.