Generating an emergency recovery key for Wrike Lock

The following commands should be carried out in Mac or Linux terminal. In order to do the same on Windows you need to install Cygwin first.

Here are the steps to follow in order to generate recovery key for Wrike Lock:

1. Create a recovery directory and open it:

```
bash-3.2$ mkdir setup_encryption_recovery
bash-3.2$ cd setup_encryption_recovery/
```

The name of the directory can be arbitrary, **setup_encryption_recovery** is a sample.

2. Generate your private key (emergency recovery key). Enter pass phrase for private key:

```
bash-3.2$ openssl genrsa -aes256 -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: ******

bash-3.2$ ls
private.pem
```

Do not forget the passphrase for your private key, it cannot be recovered. Do not share the **private.pem** file with anyone, including Wrike employees. You will use it only for recovery key decryption in case of loss of access to KMS. Save the **private.pem** file in a secure location and do not lose it under any circumstances, or recovery will not be possible. You could use HSM to store the private key.

3. Generate your public key:

```
bash-3.2$ openssl rsa -in private.pem -pubout -outform DER | base64 > public_for_wrike.txt
Enter pass phrase for private.pem: ******
writing RSA key

bash-3.2$ ls
private.pem       public_for_wrike.txt
```

Now you have a public key that we'll use to encrypt your recovery key.

4. Find the file named **public_for_wrike.txt** in the directory and send it to Wrike Support along with the encryption key.

5. Wrike Support will help you to validate that the recovery key is working.

Support agent will send you back a file named **encrypted_test.txt**. The content of the file will be something like:

```
MIIBIjANBgkqhkiG9w0BhQEFAAOCAQ8AMIIBCgKCAQEAoTJrKW2VrhsOW5EY4D92+Rp8WsE/O2kpoq2sGZlAOaTxO3YhMZxaJdxVzMnLfZdedDWgoA8WwlyBpQMHcL
jGA3yHPPMyLEFWfLokDNR8sD/CkvNc6fiJJksh9uYnbNfgswAtzHA4LvxO4h8wRmGzO7WW7HEN9c0WT1hIWNjDC6iOOcvjB1xZHzEd7/PgOmT8CNdHphsr+/NwG+FF
tVk+osbbjknpWIuQbPtJYJWiM/mU1oeHeH2avV3tSRSecx7ce6fkdlH6K99SfRqPcFseExwOXI+EcFPj1SuzTMxTtQEIWW4GlGNhfO8KeoVKS0s1u5gfAk9p3SUt+9
c9TJSuZwIDAQAB
```

Place the **encrypted_test.txt** file in the **setup_encryption_recovery** directory and validate the recovery key:

```
bash-3.2$  cat encrypted_test.txt | base64 --decode | openssl rsautl -decrypt -inkey private.pem

Enter pass phrase for private.pem: ******
Wrike recovery test
```

Getting "Wrike recovery test" in output is an expected outcome indicating that the recovery process is working.

Reference: The full list of commands

```
bash-3.2$ mkdir setup_encryption_recovery
bash-3.2$ cd setup_encryption_recovery/
bash-3.2$ openssl genrsa -aes256 -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: ******
bash-3.2$ openssl rsa -in private.pem -pubout -outform DER | base64 > public_for_wrike.txt
Enter pass phrase for private.pem: ******
writing RSA key
bash-3.2$ cat encrypted_test.txt | base64 --decode | openssl rsautl -decrypt -inkey private.pem
Enter pass phrase for private.pem: ******
Wrike recovery test
```