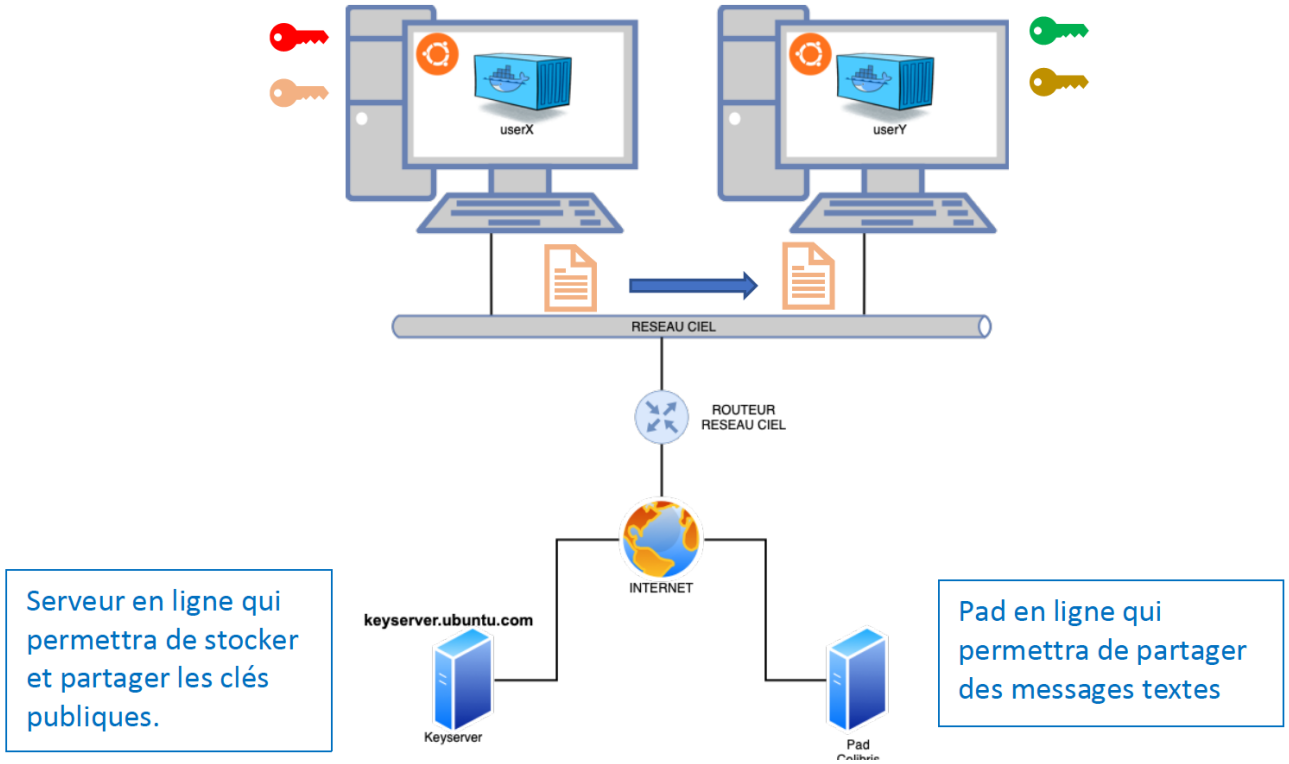
Lycée LOUIS JOUVET Accompagner pour réussir	Cybersécurité *Création d'une paire de clé privée/publique et chiffrage d'un message*	***BTS CIEL*** ***Semestre 1 2024_2025***

Objectif : dans ce TP, nous allons créer une paire de clé publique/privée sur chaque conteneur Docker. Nous allons ensuite les utiliser pour réaliser un chiffrage/déchiffrage asymétrique d'un message texte.



1. Création du conteneur ubuntu

docker run -it --name=userXouY --hostname=userXouY image

exemple : docker run -it –-name=user1 –-hostname=user1 ubuntu

2. Installation des paquets

apt update

apt install -y nano iproute2 iputils-ping gpg

3. Génération d'une paire de clé privée/publique

gpg --full-generate-key

```
root@user1:/# gpg --full-generate-key
gpg (GnuPG) 2.2.27; Copyright (C) 2021 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: directory '/root/.gnupg' created
gpg: keybox '/root/.gnupg/pubring.kbx' created
Please select what kind of key you want:
   (1) RSA and RSA (default)
   (2) DSA and Elgamal
   (3) DSA (sign only)
   (4) RSA (sign only)
  (14) Existing key from card
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (3072) 2048
Requested keysize is 2048 bits
Please specify how long the key should be valid.
         0 = key does not expire
      <n>  = key expires in n days
      <n>w = key expires in n weeks
      <n>m = key expires in n months
      <n>y = key expires in n years
Key is valid for? (0) 1
Key expires at Fri Oct  4 11:33:34 2024 UTC
Is this correct? (y/N) y

GnuPG needs to construct a user ID to identify your key.

Real name: user1
Email address: user1@btsciel.fr
Comment:
You selected this USER-ID:
    "user1 <user1@btsciel.fr>"

       Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O
```

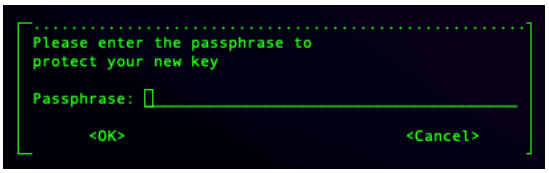
Sélectionner le type de chiffrage (1 : type de chiffrage RSA)

Choisir la taille de la clé (2048 dans notre TP)

Choisir la durée de validité de la clé (1 pour 1 jour)

Indiquer le nom et l'adresse mail

Valider le paramétrage (O)



Créer un mot de passe qui peut également être une phrase

```
pub   rsa2048 2024-10-03 [SC] [expires: 2024-10-04]
      FF8DC2B9234322D696EFD10E0B2F6A6C99A5B519
uid                      user1 <user1@btsciel.fr>
sub   rsa2048 2024-10-03 [E] [expires: 2024-10-04]
```

Une fois terminé, une paire de clé publique/privée est créée. (On visualise ici la clé publique avec son identifiant qui servira pour la suite du TP)

4. Exportation de la clé publique dans un fichier public.key

gpg --armor --export [identifiant ou adresse e-mail de la clé] > public.key

exemple : gpg --armor --export user1 > public.key

5. Transfert de la clé publique dans le keyserver

gpg --keyserver hkp://keyserver.ubuntu.com --send-keys [ID_DE_VOTRE_CLE_PUBLIQUE]

exemple : gpg --keyserver hkp://keyserver.ubuntu.com --send-keys 517C0CF301EF6528D81048CF005F24956DF77E0D

6. Téléchargement de la clé publique du voisin

gpg --keyserver hkp://keyserver.ubuntu.com --search-keys [ADRESSE_EMAIL_ASSOCIEE_A_LA_CLE_PUBLIQUE]

exemple : gpg --keyserver hkp://keyserver.ubuntu.com --search-keys user2@btsciel.fr

7. Création d'un message sur le user2

Editer un fichier texte nommé message.txt que vous compléterez avec le texte de votre choix.

nano message.txt

8. Chiffrage du message

gpg -e -r user2 -o encrypted.txt --armor message.txt

9. Transfert du message chiffré au user1

Connectez-vous sur l'outil de création de pad : https://pad.colibris-outilslibres.org

Créer un pad et y copier le contenu du message chiffré (encrypted.txt).

Copier le message de votre voisin et collez le dans un fichier nommé encrypted-voisin.txt.

10. Déchiffrage du message du voisin

gpg --decrypt encrypted-voisin.txt > message-voisin.txt

11. Lecture du message obtenu

Utiliser cat pour lire le message-voisin.txt et vérifier de déchiffrage.