AUTHENTIFICATION MULTIFACTEUR

DESCRIPTION DU THÈME

Propriétés	Description
Intitulé long	Mise en place d'une authentification multifacteur à l'aide d'une clé de sécurité matérielle ou d'un service OTP (one-time password).
Formation(s) concernée(s)	☒ BTS Services Informatiques aux Organisations
Matière(s)	☒ Bloc 3 SIO1 – Cybersécurité des services informatiques
Présentation	Après avoir abordé les limites et les faiblesses de l'authentification par mot de passe, l'objectif de cette activité est de comprendre les notions de MFA et 2FA puis d'être en mesure de les mettre en œuvre dans des contextes précis. Un dossier documentaire est mis à disposition dans le fichier « dossierdocu-MFA »
Compétences	B.3.2.2 Déployer les moyens appropriés de preuve électronique B.3.3.2 Identifier les menaces et mettre en œuvre les défenses appropriées B.3.3.3 Gérer les accès et les privilèges appropriés B.3.3.4 Vérifier l'efficacité de la protection
Prérequis	Fondamentaux concernant les systèmes d'exploitation et le réseau. Avoir abordé les limites de l'authentification par login et mot de passe. Connaître les fondamentaux en matière de cryptographie (chiffrement, signature, hachage, salage).
Outils	Machines virtuelles, clé de sécurité matérielle, smartphone avec application OTP.
Mots-clés	2fa, mfa, totp, hotp, fido2, webauthn, yubikey, mot de passe, authentification
Durée	6 heures
Auteur·e·s	Quentin Demoulière avec les précieuses recommandations et les relectures minutieuses de Florian Maury, Zakari Berremili, et Apollonie Raffalli.
Version	v 1
Date de publication	Mars 2024

DERNIÈRES RÉVISIONS

Ce tableau contient les modifications apportées au document après sa publication uniquement.

Date	Auteur·e	Description
01/2024	Q. Demoulière	Sujet version 1.0

CONTEXTE

Suite à un test d'intrusion réalisé par un prestataire et au rapport d'audit qui vous a été transmis, plusieurs mots de passe utilisateurs ont été récupérés. Cela a mis en évidence les limites de l'authentification appliquée actuellement au sein de votre entreprise. Votre responsable a pris le temps de lire les recommandations relatives à l'authentification multifacteur et aux mots de passe publiées par l'ANSSI[1]. Ainsi, il vous est demandé de proposer des solutions permettant de renforcer la sécurité liée à l'authentification.

> ⚠ Important ! Avant de commencer le TP, il est important de noter le numéro de série de votre yubikey afin de pouvoir la repérer facilement lors des prochaines séances.

PLATEFORME NÉCESSAIRE

1 yubikey 5 NFC.

1 smartphone Android avec l'application FreeOTP+ ou un iPhone avec l'application FreeOTP Authenticator.

1 machine virtuelle cliente sous Windows 10/11 Professionnel/Education.

1 machine virtuelle cliente sous Ubuntu ou Kali Linux.

1 machine virtuelle serveur sous Debian 12 avec OpenSSH installé et fonctionnel.

L'intégralité des VM doit disposer d'un accès réseau complet.

MISE EN PLACE D'UNE AUTHENTIFICATION MULTIFACTEUR SUR LE POSTE CLIENT WINDOWS 10/11 AVEC UNE YUBIKEY

> ⚠ Important ! Réaliser un snapshot de votre machine virtuelle Windows 10/11 avant de commencer ce TP afin de pouvoir revenir à son état initial avant l'installation de l'authentification 2FA[2].

Q1. Expliquer les limites de l'authentification par simple mot de passe.

Q2. Expliquer en fonction de quel(s) critère(s) définit-on le type d'authentification que l'on met en œuvre pour l'accès à un service, une application ou un système d'exploitation.

> ⚠ Attention ! La yubikey dispose de deux codes PIN, un code PIN utilisateur qui est par défaut 123456 et un code PIN administrateur 12345678. Il ne faut jamais dans le cadre des travaux pratiques modifier le code PIN administrateur sous peine de rendre la clé totalement inutilisable.

Créer un nouvel utilisateur local nommé Christian Lemale (login : clemale, mot de passe : etudiant_007, cocher « le mot de passe n'expire jamais ») et se connecter avec ce compte sur le poste Windows 10/11.

1 *https://cyber.gouv.fr/publications/recommandations-relatives-lauthentification-multifacteur-et-aux-mots-de-passe*
2 *Authentification reposant sur 2 facteurs distincts*

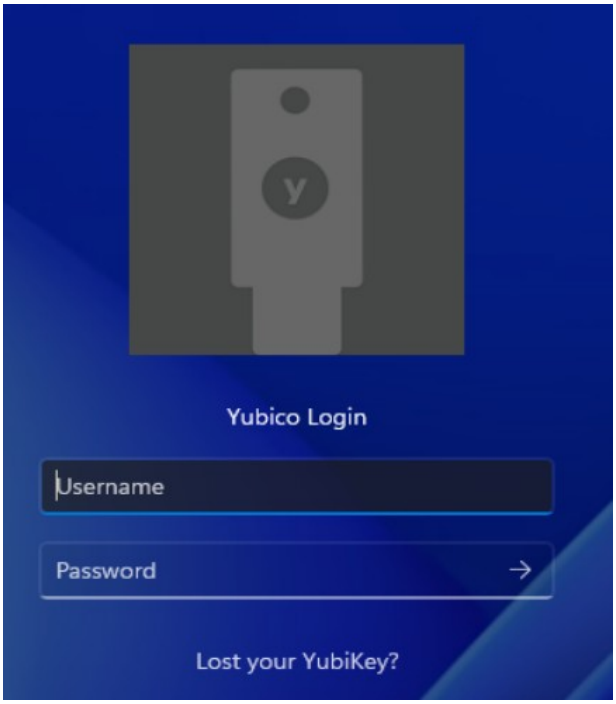
Installer le logiciel Yubico Login for Windows (une authentification en tant qu'administrateur vous sera demandée) qui vous permettra de mettre en place l'authentification multifacteur pour un compte Windows local.

https://www.yubico.com/products/computer-login-tools/

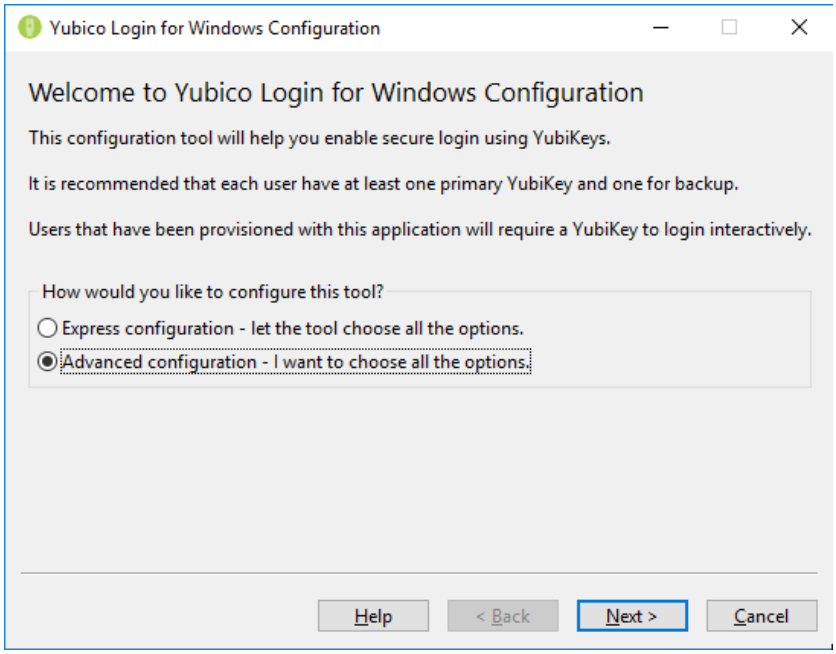
Le système redémarre.

Se connecter à nouveau à l'aide de l'utilisateur clemale.

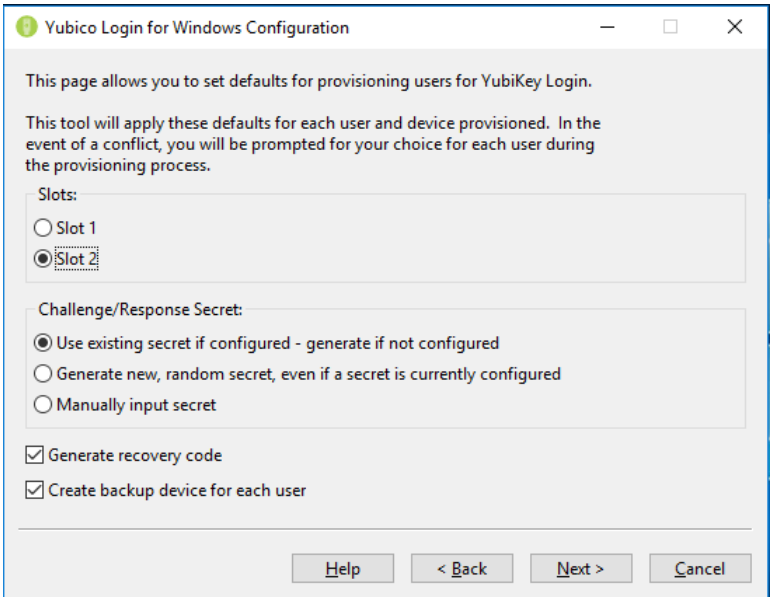
> ⚠ Attention, sur Windows 11, l'écran de login a été modifié, mais tant que la double authentification n'est pas configurée, il est toujours possible de se connecter via un login/mot de passe.



Puis rechercher l'application Login Configuration. Une authentification en tant qu'administrateur vous sera demandée.



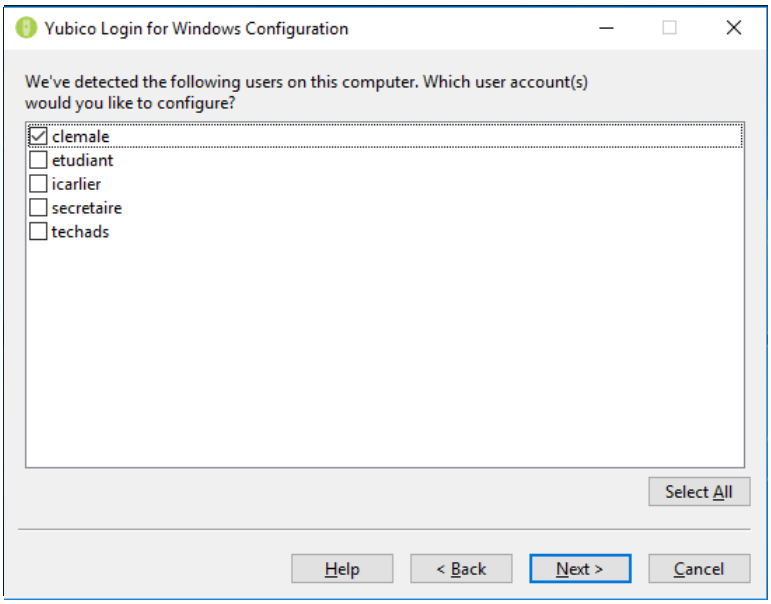
Choisir la configuration avancée afin de comprendre comment l'authentification va être configurée.



Par défaut, les yubikey disposent d'un Slot 1 préconfiguré avec une clé secrète. Dans le cas présent, nous allons donc choisir le Slot 2 qui est vide et sélectionner « Use existing secret if configured - generate if not configured ». Nous générerons un code de récupération en cas de perte ou de vol de la clé de sécurité puis nous décocherons « Create backup device for each user ».

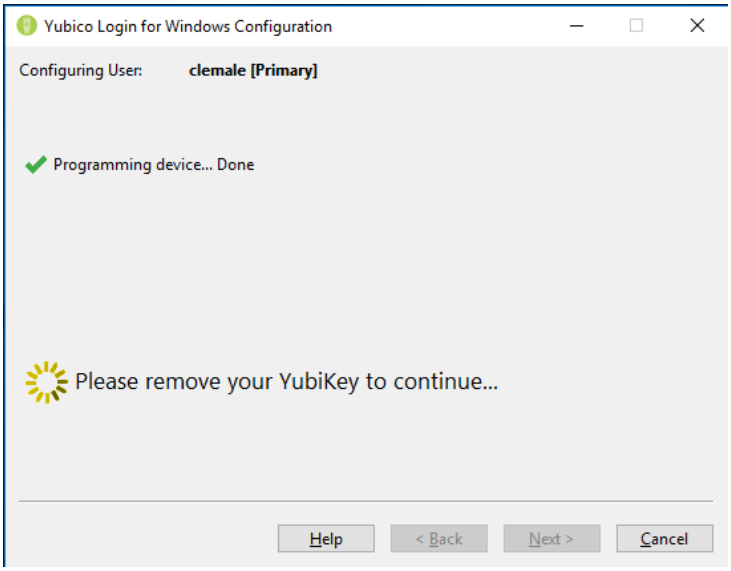
Q3. À quoi sert cette case ? Pourquoi est-elle importante et pourquoi la décocher malgré tout dans le cadre de cette activité précise ?

Choisir ensuite les utilisateurs qui seront concernés par la mise en place d'une authentification multifacteur.

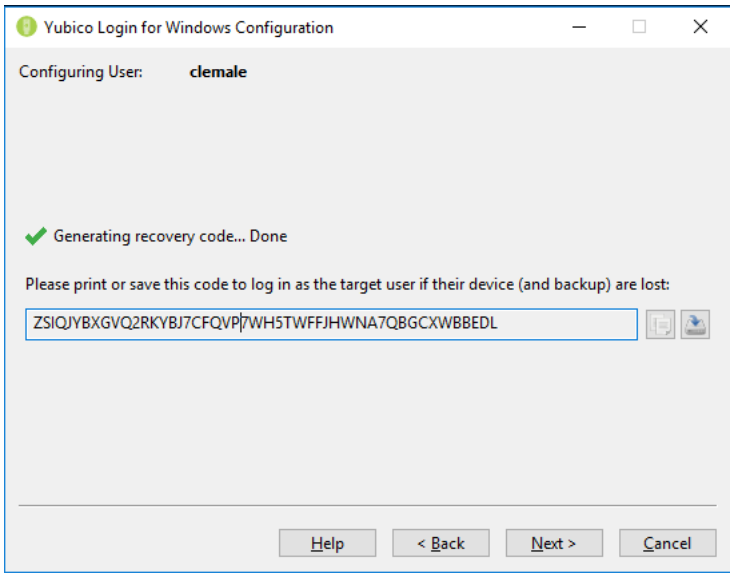


Suivre ensuite les instructions. Connecter en USB la yubikey (sur Linux, s'assurer que l'utilisateur est bien rattaché au group « vboxusers ») puis l'enlever lorsque l'application le demande.

> ⚠ Attention ! Puisque nous utilisons une machine virtuelle, il est indispensable de permettre la détection de la clé de sécurité physique USB-A sur la VM et non sur la machine hôte. Pour cela dans la barre de menu de la machine virtuelle, cliquer sur Périphériques > USB > Yubico Yubikey. Cocher la case pour activer la détection de la clé matérielle par la VM.

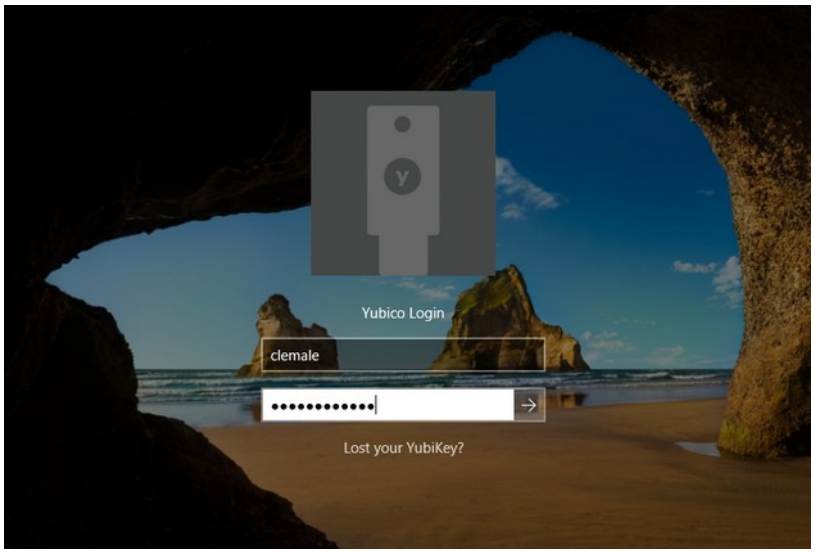


Q4. À quoi sert le code affiché à l'écran ?



Q5. Proposer une solution permettant de conserver ce code de sécurité en assurant une forte confidentialité.

Se déconnecter de la session, puis essayer de se connecter avec le compte clemale à l'aide du mot de passe simple puis dans un second temps du mot de passe et de la clé de sécurité.



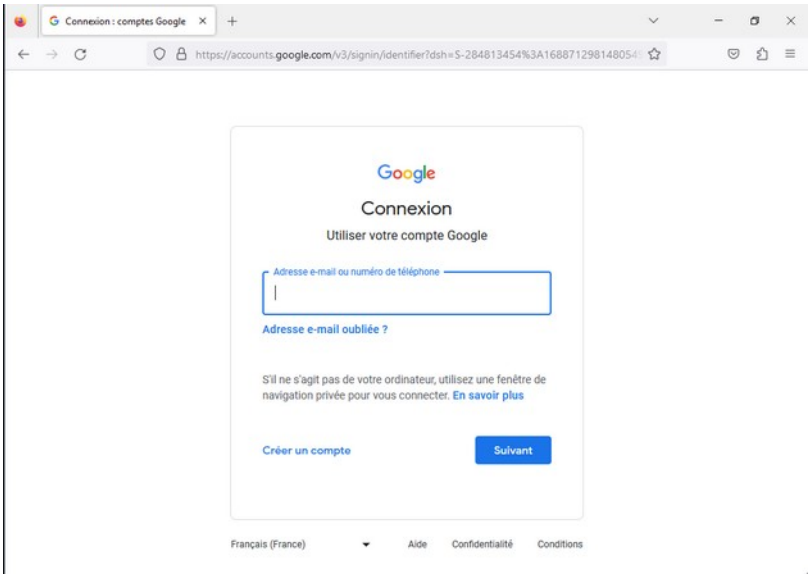
Q6. Déterminer les catégories de facteur mobilisées lors de cette phase d'authentification.

Q7. À l'aide du dossier documentaire, déterminer quel mécanisme d'authentification est utilisé entre la clé de sécurité et le système Windows 10/11. Puis proposer un schéma détaillant les grands principes de fonctionnement de la méthode d'authentification sélectionnée.

AUTHENTIFICATION MULTIFACTEUR SUR UNE APPLICATION WEB

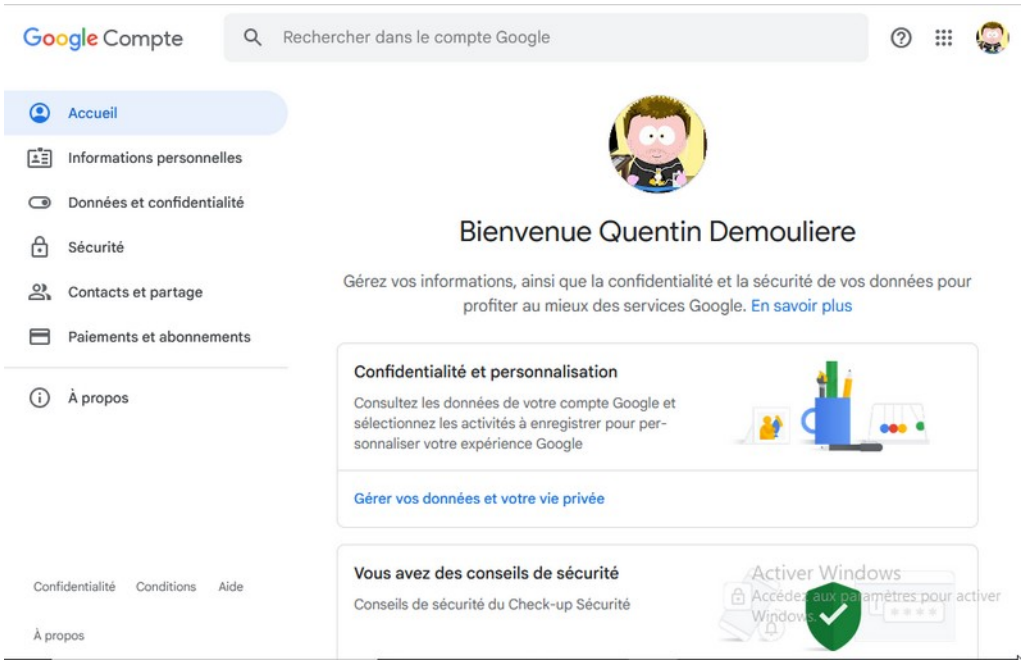
L'accès à certaines applications web considérées comme sensibles pour l'entreprise nécessitent également la mise en place d'une authentification MFA. Dans le cadre de cette activité, il vous est demandé de la mettre en œuvre temporairement sur votre compte personnel Google (le créer le cas échéant).

Installer le navigateur web Mozilla Firefox sur le PC client depuis le compte clemale.

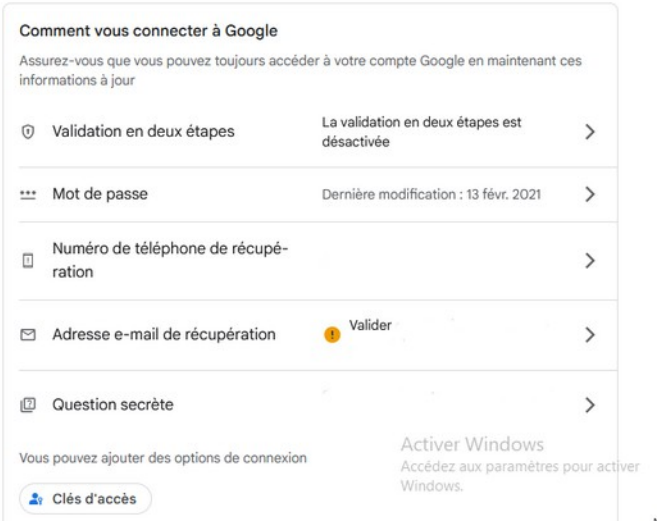


Se connecter sur la page d'authentification Google.

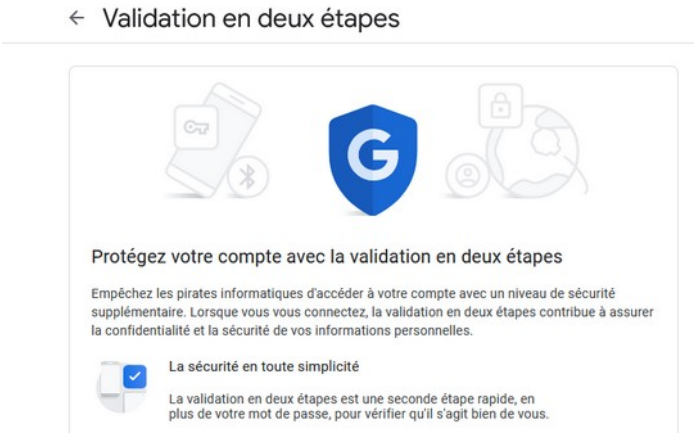
Une fois authentifié, cliquer sur Gérer votre compte Google puis sur le menu Sécurité.

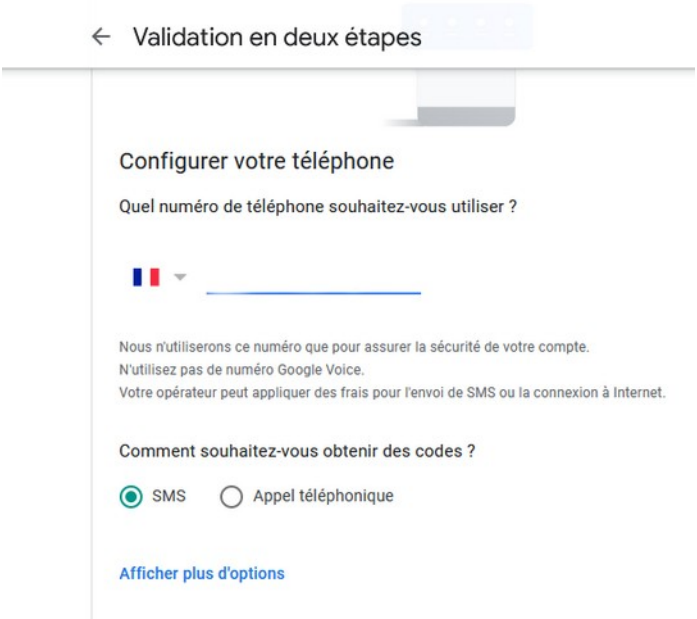


Cliquer ensuite sur « Validation en deux étapes » puis sur « Commencer ».



⚠ Attention ! Si vous avez déjà configuré une authentification double-facteur auparavant, il est nécessaire de la désactiver afin de pouvoir ajouter la yubikey.



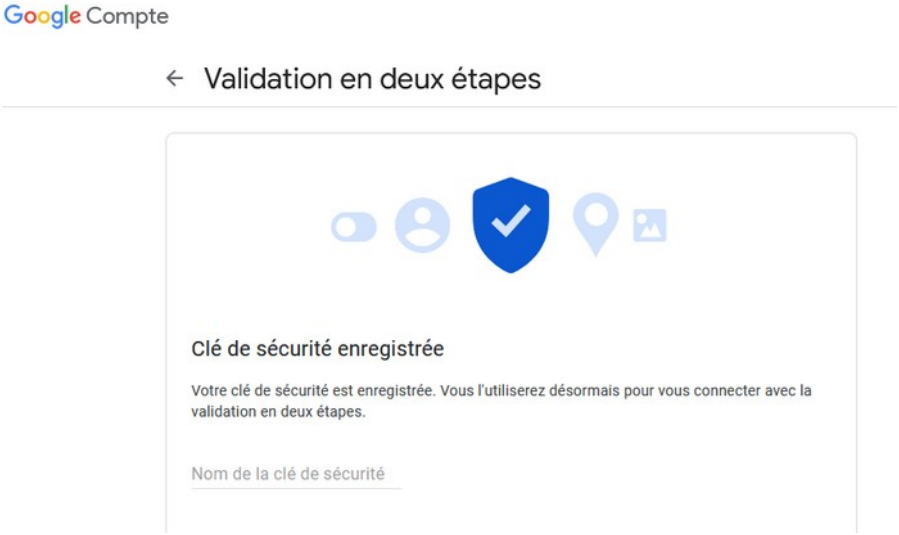


La configuration par défaut proposée est l'utilisation d'une authentification MFA en envoyant un code à usage unique (OTP) par SMS. Dans le cas présent, il vous est demandé d'utiliser une clé de sécurité.

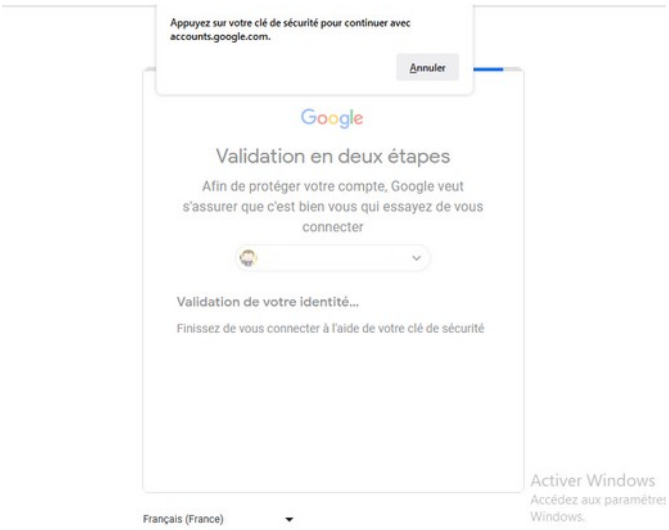
Q8. Pourquoi la configuration par défaut proposée par Google renforce-t-elle la sécurité de l'authentification par rapport à l'utilisation d'un simple mot de passe ? Pourquoi s'avère-t-elle par contre moins efficace que l'utilisation d'une clé de sécurité physique ?

Cliquer sur « Afficher plus d'options » puis « Clé de sécurité ». L'étape suivante consiste donc à insérer la clé de sécurité dans le port USB puis à l'enregistrer.

Cliquer sur « Continuer » (en ne tenant pas compte de l'avertissement émis par le navigateur web Firefox).

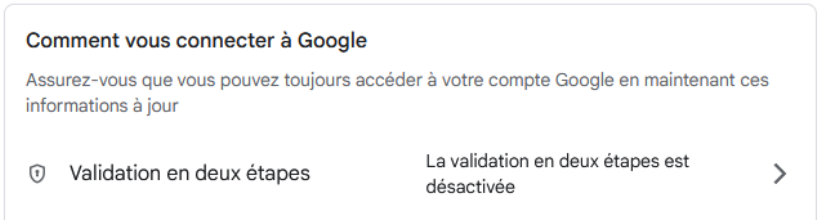


La clé de sécurité et l'authentification en deux étapes sont maintenant correctement configurées. Se déconnecter du compte Google en cours d'utilisation puis essayer de se connecter à nouveau avec et sans la yubikey.



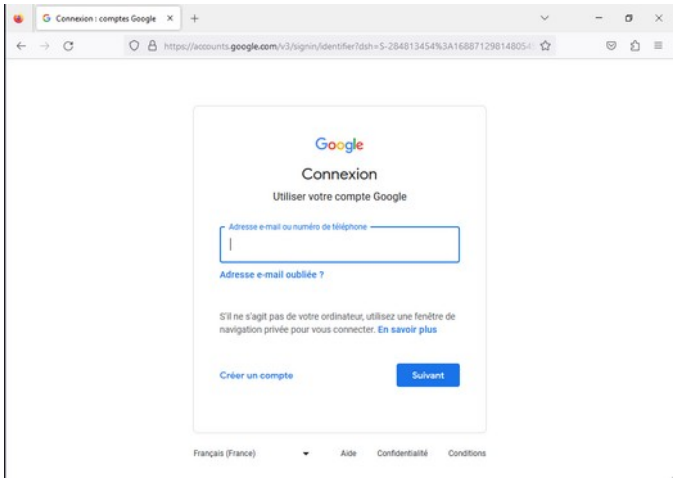
Q9. À l'aide du dossier documentaire et de recherches personnelles, définir quelle méthode d'authentification MFA est utilisée dans le cas présent.

Une fois l'activité terminée, désactiver l'authentification MFA sur le compte google utilisé. Cliquer sur « Sécurité > Validation en deux étapes > Clé de sécurité (Par défaut) > Supprimer (X) ».



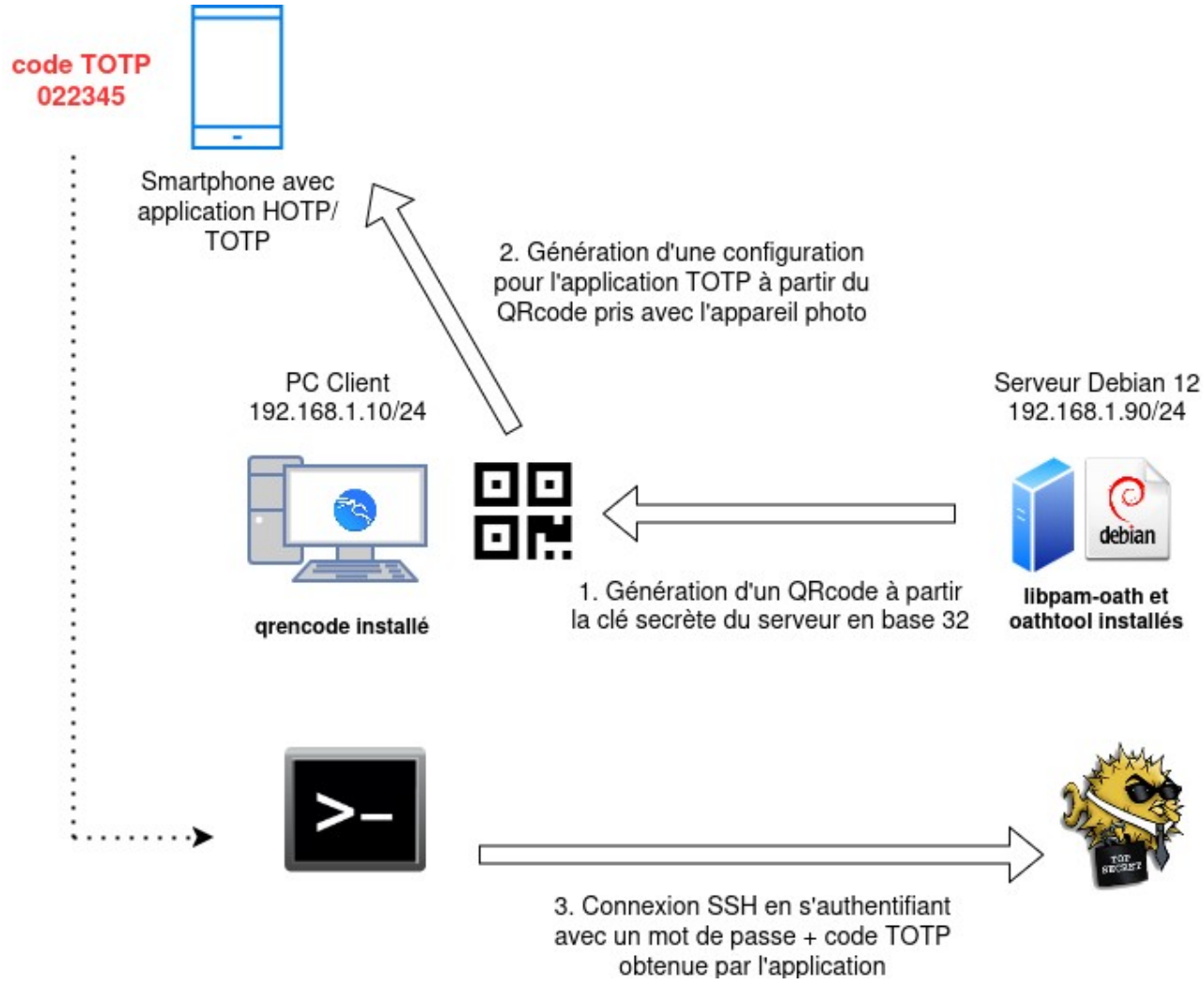
S'assurer que la validation en deux étapes est dorénavant désactivée.

Se déconnecter du compte et vérifier que l'authentification par simple mot de passe est de nouveau opérationnel.



Ranger la clé de sécurité dans son étui et la remettre dans la boite commune.

MISE EN PLACE D'UNE AUTHENTIFICATION SSH 2FA AVEC UN MOT DE PASSE ET OTP SUR UN SERVEUR DEBIAN



Présentation de l'infrastructure

Prérequis sur le smartphone et la machine cliente

Sur votre smartphone, installez l'application FreeOTP+ sur Android ou FreeOTP Authenticator sur IOS[3]. Sur votre poste client Kali ou Ubuntu, installez le paquet qrencode qui permet de générer un QR code afin de paramétrer l'application FreeOTP+ en lien avec le service oath présent sur le serveur Debian 12.

```
etudiant@client:~$ sudo apt install qrencode
```

On s'assurera également que la connexion SSH entre le client et le serveur Debian est pleinement opérationnelle (le nom d'utilisateur doit être éventuellement adapté). Si ce n'est pas le cas, il sera impératif d'installer le service OpenSSH sur le serveur Debian 12 (sudo apt install openssh-server).

```
etudiant@client:~$ ssh etudiant@192.168.1.90
```

3 *N'importe quelle application OTP compatible HOTP/TOTP peut être utilisée*

```
etudiant@192.168.1.90's password:
Linux serveur 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-
30) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan  3 19:34:10 2024 from 192.168.1.85
etudiant@serveur:~$
```

Configuration du serveur

Dans un premier temps, il sera nécessaire d'installer les paquets permettant de mettre en œuvre le mécanisme d'OTP nommé OATH[4] sur le serveur.

```
etudiant@serveur:~$ sudo apt install libpam-oath oathtool
```

Puis, nous allons définir un secret sous forme hexadécimal qui sera utilisé par le générateur TOTP/HOTP en lien avec l'utilisateur étudiant et qui sera soumis à la double authentification.

> ⚠ Attention ! Ce secret doit être jalousement gardé, car c'est la clé de voûte servant à la génération des mots de passe à usage unique. S'il est compromis, c'est l'ensemble de l'authentification OTP qui sera impacté.

```
etudiant@serveur:~$ sudo -i
root@serveur:~# KEY=$(openssl rand -hex 20)
root@serveur:~# echo "HOTP/T30/6 etudiant - ${KEY}" >> /etc/security/users.oath
root@serveur:~# chown root /etc/security/users.oath
root@serveur:~# chown 600 /etc/security/users.oath
```

Q10. Expliquez dans le détail, ce que fait chacune de ces commandes.

Nous allons maintenant configurer PAM (Pluggable Authentication Modules), le service qui contrôle les authentifications sur le serveur Debian.

```
root@serveur:~# nano /etc/pam.d/sshd
```

```
# PAM configuration for the Secure Shell service

# Standard Un*x authentication.
#@include common-auth

auth required pam_unix.so nullok_secure
auth required pam_oath.so usersfile=/etc/security/users.oath window=20 digits=6
```

Nous commentons la ligne @include common-auth car elle empêche l'authentification OTP même en cas de connexion avec le bon mot de passe.

La ligne suivante impose l'authentification par mot de passe stocké en local sur le système et interdit les mots de passe vides.

4 *Initiative pour l'authentification ouverte est une collaboration industrielle visant à développer une architecture de référence ouverte utilisant des normes ouvertes pour promouvoir l'adoption d'une authentification plus robuste que le simple mot de passe*

La dernière impose une fois l'authentification par mot de passe réussie, une deuxième authentification par OTP. Nous faisons référence au fichier contenant le (ou les) nom d'utilisateur concerné ainsi que le secret servant à générer des mots de passe à usage unique. Ce mot de passe disposera de 6 chiffres et il sera possible de générer 20 codes à usage unique valides.

Il nous reste maintenant à éditer le fichier de configuration du service SSH afin de définir l'usage de l'authentification 2FA.

```
root@serveur:~# nano /etc/ssh/sshd_config
```

```
ChallengeResponseAuthentication yes
#KbdInteractiveAuthentication no
#La ligne ci-dessous est normalement déjà décommentée
UsePAM yes
```

Nous nous assurons de commenter la ligne KdbInteractiveAuthentication et d'avoir les deux autres lignes activées avec la valeur yes. Nous pouvons ensuite redémarrer le service SSH.

```
root@serveur:~# systemctl restart ssh
```

Enfin, il nous reste à récupérer le secret en base 32 qui nous permettra ensuite de générer sur le poste client un QR code pour notre application Android.

```
root@serveur:~# cat /etc/security/users.oath
HOTP/T30/6 etudiant - 65f43c705ce51c9c058ec8bb4b7f64b656681866
root@serveur:~# oathtool -v -d 6 65f43c705ce51c9c058ec8bb4b7f64b656681866
Hex secret: 65f43c705ce51c9c058ec8bb4b7f64b656681866
Base32 secret: MX2DY4C44UOJYBMOZC5UW73EWZLGQGDG
Digits: 6
Window size: 0
Start counter: 0x0 (0)
```

> ⚠ Attention ! Pour que la partie TOTP soit pleinement fonctionnelle, vous devez vous assurer que les horloges des différentes machines sont synchronisées et à l'heure.

Configuration du client et de l'application Android FreeOTP+

Il s'agit maintenant de paramétrer correctement la machine cliente et l'application Android FreeOTP+ afin de rendre opérationnel l'authentification SSH 2FA.

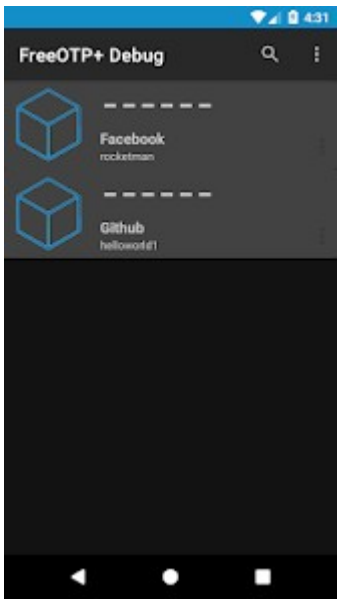
Nous allons d'abord générer sur le poste client (Ubuntu ou Kali) un fichier png contenant un QR code que nous soumettrons à l'application FreeOTP+.

```
etudiant@client:~$ qrencode -o etudiant.png
'otpauth://totp/etudiant@192.168.1.90?secret=MX2DY4C44UOJYBMOZC5UW73EWZLGQGDG'
```

```
etudiant@client:~$ ls -l
…
-rw-r--r-- 1 etudiant etudiant        471  3 janv. 23:08  etudiant.png
```

Nous pouvons ouvrir ce fichier PNG sur le poste client puis ouvrir l'application FreeOTP+ sur le smartphone.

Sélectionner l'icône « Appareil photo » en bas à droite. Puis prenez en photo le QR code présent sur l'écran du client. Une nouvelle configuration pour votre utilisateur et votre serveur est automatiquement créée.



Nous pouvons ensuite supprimer le fichier PNG contenant le QR code, car il contient le secret à ne pas compromettre.

Sur le client, nous pouvons lancer une connexion SSH vers le serveur avec le compte etudiant. Après avoir entré votre mot de passe, un OTP vous est demandé. Dans l'application FreeOTP+, sélectionnez la nouvelle configuration. Celle-ci vous fournit un code de 6 chiffres valable 30 secondes.

```
etudiant@client:~$ ssh etudiant@192.168.1.90
(etudiant@192.168.1.90) Password:
(etudiant@192.168.1.90) One-time password (OATH) for `etudiant':
Linux serveur 6.1.0-17-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.69-1 (2023-12-
30) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan  3 22:17:13 2024 from 192.168.1.85
etudiant@serveur:~$
```

Q11. Expliquer quel est le type d'authentification utilisé ici.

Q12. Expliquer en quoi cette solution est moins sécurisée que l'utilisation de FIDO 2.

Le mot de passe reste le maillon faible de l'authentification.

Q13. Expliquer en quoi l'authentification au serveur SSH au moyen d'une paire de clés privée/publique et d'un code à usage unique TOTP est plus sécurisée ?

Pour aller plus loin

Mettre en place l'une de ces propositions (paire de clés SSH + TOTP ou FIDO 2) et réaliser les tests nécessaires permettant de valider sa mise en œuvre effective.