Challenge de noël

Ce challenge s'effectuera du 12 décembre 2022 au 6 janvier 2023 minuit

Attention
Vous devez fournir une ou plusieurs applications pour chaque mission (pour chaque mission il faut rendre un fichier compressé en ZIP), toute modification d'une application lors d'une nouvelle mission correspondra à une nouvelle application

Ne pas oublier de commenter vos programmes

Pour chaque mission vous devrez joindre un rapport expliquant votre programme et vos choix

Penser d'abord à travailler en brut (interface de commande) avant de penser à l'IHM (interface graphique)

Pour la mission 3 du fait de la durée du défi vous n'aurez pas forcément le temps de traiter des éléments très complexe, vous devrez alors peut être inclure dans vos programmes la possibilité d'un arrêt automatique après un temps donnée et la possibilité de forcer l'arrêt. Vous pouvez aussi effectuer des tests automatiques chez vous (pendant la nuit, le WE ou vacances).

Ce défi permettra d'évaluer les compétences suivantes :
Bloc 3

- Sécuriser les équipements et les usages des utilisateurs
- Vérifier l'efficacité de la protection
- Participer à la vérification des éléments contribuant à la qualité d'un développement informatique
- Prendre en compte la sécurité dans un projet de développement d'une solution applicative
- Mettre en œuvre et vérifier la conformité d'une solution applicative et de son développement à un référentiel, une norme ou un standard de sécurité

Bloc 2

- Analyser un besoin exprimé et son contexte juridique
- Participer à la conception de l'architecture d'une solution applicative
- Modéliser une solution applicative
- Identifier, développer, utiliser ou adapter des composants logiciels

Ce challenge vous permettra de comprendre l'utilité du choix d'un bon mot de passe.
Bien que ce challenge soit plus orienté développement, il peut être demandé aux personnes travaillant sur des systèmes de réaliser de telle opération.

Notation
Attention il est conseillé de rendre chaque mission dès que celle-ci est réalisé.

- La première mission est notée sur 5
- La deuxième mission est notée sur 10
- La troisième mission est notée sur 2
- La quatrième et cinquième mission est notée sur 1

Divers bonus seront attribués pour ce TP :

- Bonus de 1 point pour toute personnes ayant rendu l'ensemble du challenge
- Pour les missions 1 et 2 la notes est majoré de 25 % pour les 2 premiers de chaque groupe ayant, de 20 % pour les 3 et 4 ème et 10% pour le 5 et 6 ème (attention les programmes devront reprendre l'ensemble des demandes et être fonctionnelle)
- Pour la troisième, quatrième et cinquième mission la notes est majoré de 25 % pour les 5 premières des 2 classes

I. Contexte :

Suite à une attaque informatique sur ses serveurs, l'entreprise Medinov désire vérifier que les différents mots de passe utilisé au sein de son entreprise sont suffisamment robustes, après une recherche, elle n'a pas trouvé d'outil suffisamment efficace et pertinent pour répondre à ses besoins.

Elle a donc décidé d'embaucher un développeur spécialisé en cybersécurité pour réaliser les outils qui répondrai à ses besoins.

Après avoir passé une annonce pour cette offre d'emploi, et avoir passé divers entretiens, elle se trouve dans une impasse car environ 60 candidats sembleraient avoir le profil idéal.

La société Medinov décide donc de sélectionner le candidat sur ses réelles compétences en pratique et pour celle elle organise un challenge qui lui permettra de trouver le candidat de ses rêves.

Vous faite partie des candidats retenu pour passer cette épreuve et vous disposer de 5 semaines (du 7/12/2022 au 11/01/2023) pour rendre vos travaux.

Toutes les personnes ayant participer à ce challenge seront récompensé par rapport aux travaux réalisé, mais aussi par rapport à la qualité des documents et informations fourni avec les applications réalisées.

II. Mission 1

Créer en python une application permettant de chiffrer les mots de passe de l'utilisateur

L'application devra comprendre

- Saisie par l'utilisateur d'un mot de passe
- un hachage à l'aide de bcrypt
- Une mesure de la force des mots de passe (penser à utiliser les expression régulière), indication de la valeur au fur et à mesure de la saisie du mot de passe
- Possibilité d'ajouter un salage (choisi par l'utilisateur ou généré par l'application)

III. Mission 2

Créer en python une application permettant de craquer les mots de passe haché à l'aide de bcrypt
L'application devra permettre :

- La force brute
- L'utilisation de dictionnaire
- Mélange de la force brut et du dictionnaire
- Remplacement de caractère par d'autre ou d'une lettre par sa position dans l'alphabet
- Ajout de lettre
- Par l'utilisation de majuscule
- Utilisation de salage

IV. Mission 3

Pour permettre de tester votre application réaliser un ensemble de test permettant d'établir un tableau et graphique permettant de déterminer la résistant moyenne des mots de passe selon la taille et la composition de celui-ci.
Pour cela vous devrez

- Penser à utiliser soit un timer ou un calcul de temps
- Effectuer des tests en automatique avec sortie d'un rapport (modification des programmes précédent)
- Les tests doivent s'effectuer sur les possibilités d'attaque proposée lors de la mission précédente (chaque type d'attaque donnant une analyse différente)
    - attaque par dictionnaire

- attaque force brut avec seulement des caractères alpha en minuscule
- attaque force brut avec seulement des caractères alpha en minuscules et majuscules
- attaque force brut avec seulement des caractères alphanumériques en minuscule
- attaque force brut avec seulement des caractères alphanumériques en minuscule et majuscules
- attaque force brut avec seulement des caractères alphanumériques et des signes en minuscule
- attaque force brut avec seulement des caractères alphanumériques et des signes en minuscule et majuscule
- attaque force brut avec seulement des caractères alpha en minuscule et le remplacement de certaine lettre par une correspondance ou leur numéro dans l'alphabet
- attaque force brut sur toutes les caractéristiques ci-dessus

V. Mission 4

Proposer une solution ou méthode qui permettrait d'accélérer la vitesse de traitement de l'attaque par catalogue

Proposer un ajout à votre programme pour créer votre propre catalogue en utilisant les missions précédentes et la première question de la mission 4

VI. Mission 5

Que pensez-vous de votre logiciel pour craquer les mots de passe, pouvez-vous trouver son défaut principal et proposer les modifications à effectuer (vous n'êtes pas obligé de modifier votre code, juste une proposition de modification avec exemple suffit)

VII. Aide

Pour le calcul de la force d'un mot de passe :

https://www.ssi.gouv.fr/administration/precautions-elementaires/calculer-la-force-dun-mot-de-passe/

https://www.undernews.fr/nos-services/tester-la-force-de-votre-mot-de-passe

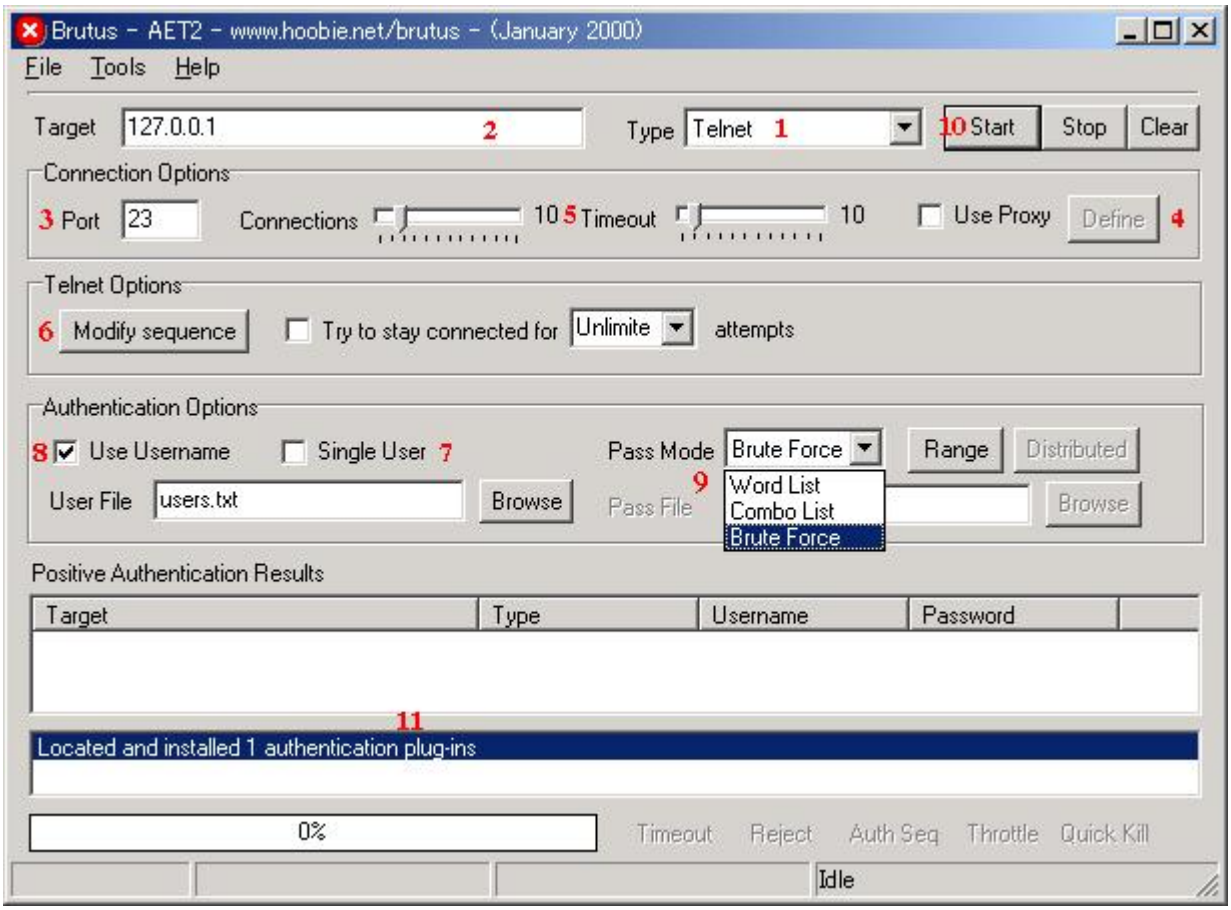
Comprendre le salage
https://apprendre-php.com/tutoriels/tutoriel-35-scuriser-les-mots-de-passe-avec-les-hashs-et-les-salts.html

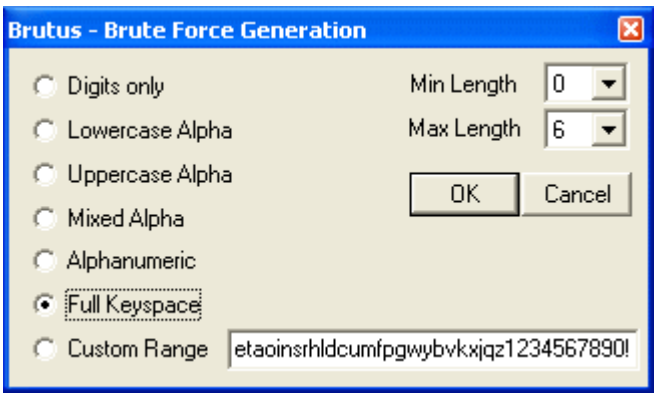
Liste des 2000 mots de passe les plus utilisés
https://github.com/tarraschk/richelieu/blob/master/french_passwords_top20000.txt

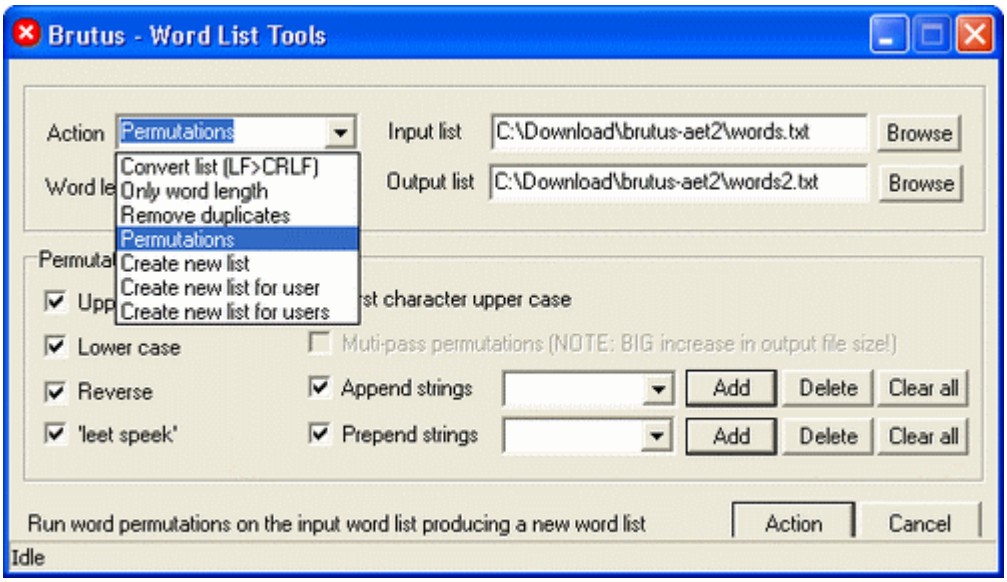
Tableau d'aide pour le remplacement de caractère

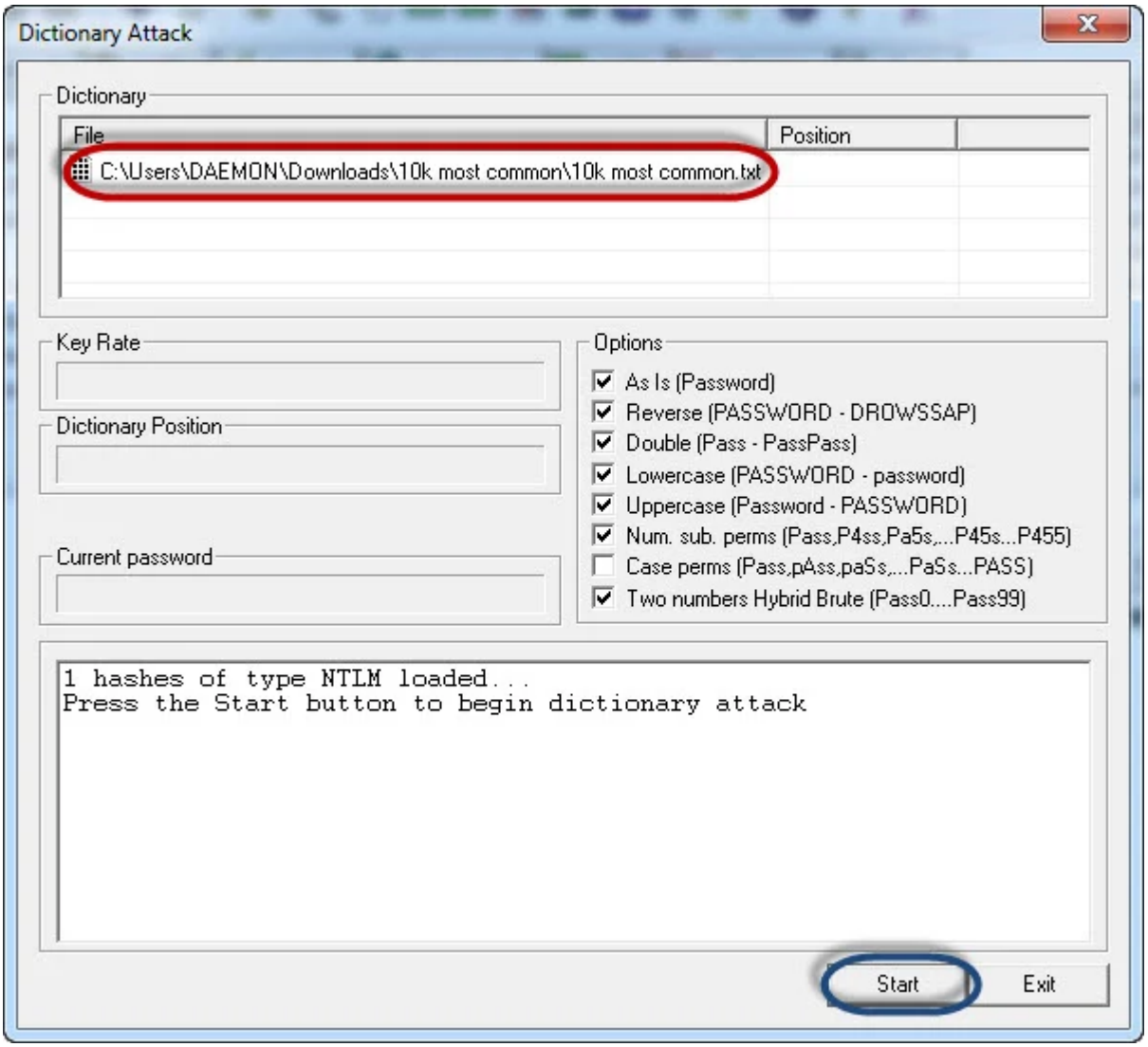
Lettre origine	a	e	o	i	l	s	b	s	et	b	g
Lettre remplacement	@	€	0	1	1	5	8	$	&	6	9

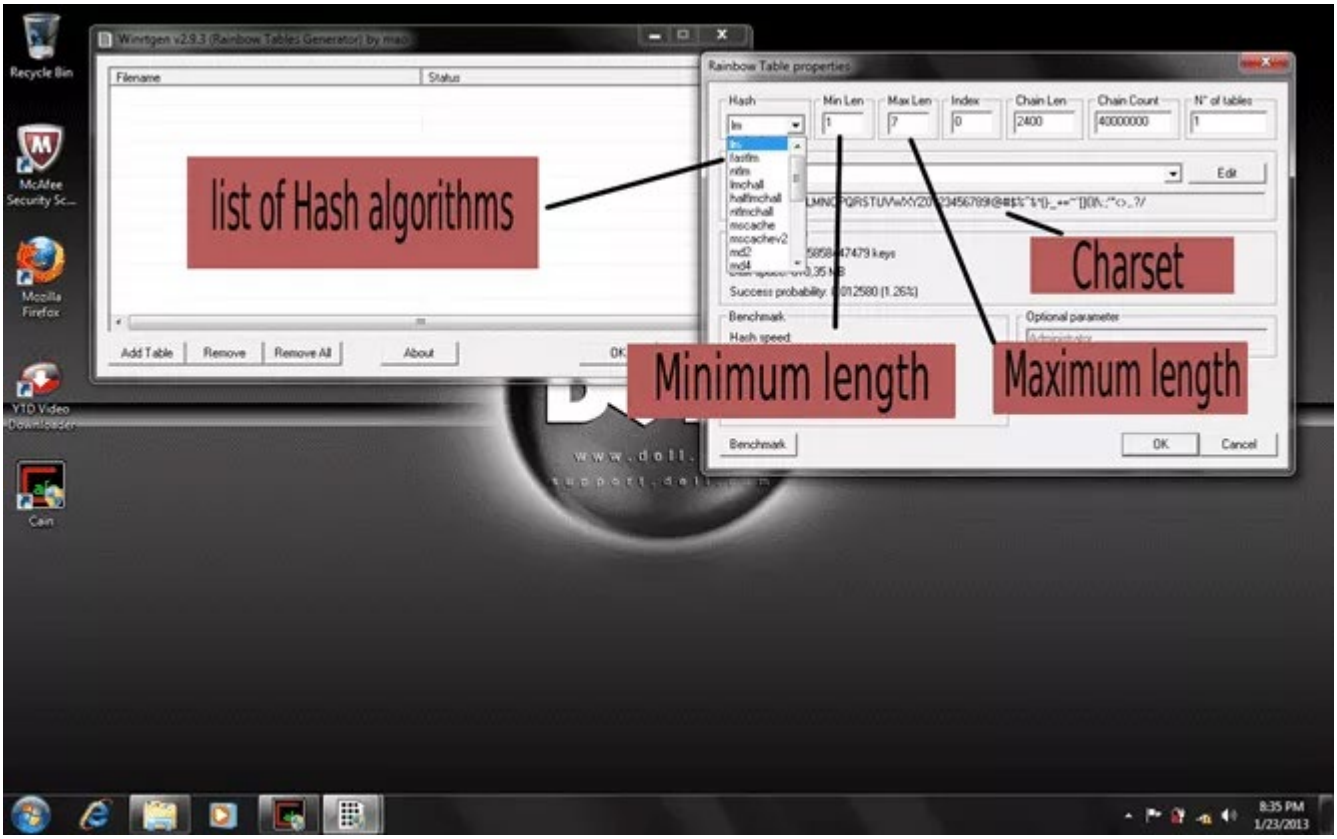
Écran de logiciel similaire








Dictionary Attack
Dictionary
File
Position
C:\Users\DAEMON\Downloads\10k most common\10k most common.txt
Key Rate
Dictionary Position
Current password
Options
As Is (Password)
Reverse (PASSWORD - DROWSSAP)
Double (Pass - PassPass)
Lowercase (PASSWORD - password)
Uppercase (Password - PASSWORD)
Num. sub. perms (Pass,P4ss,Pa5s,...P45s...P455)
Case perms (Pass,pAss,paSs,...PaSs...PASS)
Two numbers Hybrid Brute (Pass0....Pass99)
1 hashes of type NTLM loaded...
Press the Start button to begin dictionary attack
Start
Exit


list of Hash algorithms
Charset
Minimum length
Maximum length