GostCrypt

Alexis Lefrançois
Adrien Burel

November 2016

1 Présentation du projet

GostCrypt est un logiciel de chiffrement de données basé sur TrueCrypt. Ce logiciel est multi-plateforme, il permet de créer une partition chiffrée où l'on pourra par la suite placer ses dossiers sensibles. Ces partitions peuvent être déposées sur un clé USB, un disque dur ou même être envoyées par mail. il est également possible de chiffrer un système d'exploitation entièrement afin d'étendre la protection.
Libre et gratuit, le projet a été initié en 2013 par E. Filiol au laboratoire de Cryptologie et Virologie Opérationnelles du campus de Laval de l'ESIEA. Il est depuis maintenu et développé par des groupes d'étudiants de 5ème année, Paul AMICELLI, Jonathan THIEULEUX, Tristan BERTIN, Guillaume SWAENEPOEL et maintenant Alexis LEFRANÇOIS et Adrien BUREL.
Nous travaillons donc à l'évolution de ce logiciel. Le but étant d'apporter des améliorations au code pour une compréhension rapide, d'enlever du code inutilisé, de sécuriser les appels aux fonctions et d'ajouter une interface en QT.

2 Tâches assignées au projet

2.1 Compréhension et documentation du programme

2.1.1 Compréhension du programme

Comprendre la structure du programme est indispensable pour se permettre d'optimiser le code. Il est donc nécessaire de comprendre quelle fonction est appelée pour lancer le programme, quelle fonction s'occupe de l'affichage de la fenêtre, quelle fonction s'occupe d'interagir avec l'utilisateur, etc.

2.1.2 Doxygène

Il est très important afin de rédiger une documentation complète et utilisable par les personnes qui voudraient utiliser ou modifier ce projet, il est très peu commenté et assez complexe malgrès des noms de fonctions bien choisis.
Une documentation assisté par doxygène permettra d'avoir une documentation dans le code et du code lisible afin de comprendre en profondeur ce que chaque fonction fait. Cela permettra aussi la réutilisation simple de certaines fonctions lors de l'ajouts de fonctionnalitées.

2.1.3 Diagramme de fonctionnement

Pour faciliter la compréhension rapide, nous allons réaliser un diagramme de fonctionnement du projet, indiquant les fonctions qui sont appelées et à quoi elles servent.
Pour la reprise du projet par d'autres personnes il faudra donc commencer par lire ce diagramme afin de comprendre les plus grosses parties du projet et ainsi selectionner au mieux la partie qui les intéressent.

2.2 Suppression de code inutile

Quelques parties dans le code ne sont plus maintenues dans cette nouvelle version de GostCrypt. Certaines fonctions aussi ne sont pas marquées comme périmées alors qu'elles ne sont pas utilisé. il faut donc retirer ces fonctions afin de rendre le code plus court et lisible.

2.3 Détecter et recoder l'interface en Qt

Afin de pouvoir recoder l'interface en Qt il est important de bien définir les parties gérant l'ancienne interface pour ne pas affecter les parties importantes.
Le but de recoder toute l'interface en Qt permettant de rendre le projet plus portable et ainsi ne plus à avoir d'installation requise pour utiliser celui-ci, ce qui pourrait en effet permettre de lancer le logiciel depuis une clé USB pour chiffrer des informations sensibles sans avoir besoin d'installer quoi que ce soit.

2.4 Analyse du code

Faire une analyse du code en statique et en dynamique afin de vérifier que celui-ci ne comprend pas de failles exploitable.

2.5 s'informer sur l'algorithme de chiffrement

VeraCrypt qui permettait de sélectionner l'algorithme GOST pour chiffrer les données vient de se voir mis à jour et ne propose plus cette option. Après un audit du code il a été détecté que l'algorithme n'était pas totalement sûr. il était donc nécessaire de revoir la solution et de choisir un nouvel algorithme pour le chiffrement.

2.6 Ajout de fonctionnalités

L'ajout de certaines fonctionnalités permettrait d'avoir un produit plus intéressant à l'utilisation et aussi surement plus agréable.

2.7 Ecriture du rapport

l'écriture d'un rapport qui ressemblera à un mini mémoire que l'on devra rendre à la fin de notre stage de fin d'études.

3 Assignation des tâches et réalisations

3.1 Compréhension du programme

Alexis et Adrien: Nous étudions le code ensemble car il est nécessaire de bien comprendre le code pour pouvoir yu apporter des modifications. Le programme est divisé en 6 sous projets : Mount, Setup, Boot, Crypto, Format et Driver. Le sous projet Setup permet de créer l'installateur pour GostCrypt. le sous projet Mount permet quant à lui de gérer les boutons de l'interface de GostCrypt et de monter un disque.

3.2 Doxygène

Alexis : réalisation d'un script en python permettant d'écrire la base de la syntaxe utilisé pour doxygène. Il est par exemple possible d'écrire les commentaire nécessaire pour détailler les fichiers (nom de fichier) ainsi que pour les fonctions (nom de la fonction, arguments et retour de la fonction).
Adrien et Alexis : Lecture du code et écriture des commentaires (brief / return) sur les fonctions.

3.3 Diagramme de fonctionnement

Adrien : réalisation de diagrammes montrant le fonctionnement des sous projets Setup et Mount de Gostcrypt

3.4 Suppression de code inutile

Lors de l'étape de compréhension du code, nous avons déjà repérer du code non utilisé ou obsolète. Nous avons donc pris note de l'emplacement de ces fonctions, mais n'avons pas encore apporté de modifications pour le moment.

3.5 Détecter les parties d'interfaces

Alexis et Adrien : le ficher mount.c sous le projet mount contient des parties d'interface gérant le lancement du programme.
Alexis et Adrien: le fichier Setup.c sous le projet Setup contient l'interface utilisateur pour l'installation, si le projet est porté en portable ce sera une partie à effacer.
Autres interfaces utilisateur en recherchent.

3.6 interface Qt

Non commencé

3.7 Analyse du code

Non commencé

3.8 Gost revoir l'algorithme

Nécessaire puisque nous avons reçu l'information que Veracrypt avait changé sa version pour changer l'algorithme qui était mal implémenté.

4 Productions et avancement général

La compréhension du code et du programme en entier prend beaucoup de temps, il est parfois difficile de bien comprendre ce que certaines fonctions font. Actuellement nous sommes toujours dans la lecture du code en essayant de comprendre le fonctionnement global et la structure du code tout en rajoutant des commentaires dans celui-ci et produisant des schémas afin d'en avoir une meilleure vision.
les productions :
Programme python permettant d'avoir une base doxygène pour la plupart des fonctions / fichiers d'un projet C/C++.
un code doxygène comprennant des commentaires sur les plus grosses fonctions d'interface graphique des sous projets : Mount et Setup de la solution.
Un tout début de rapport.
Ce rapport intermédiaire.