Atelier : Utiliser des algorithmes de chiffrement classiques et modernes

Objectifs

Partie 1 : utiliser un algorithme de chiffrement classique

Partie 2 : utilisez un algorithme de chiffrement symétrique moderne

Partie 3 : utiliser un algorithme de chiffrement asymétrique moderne

Contexte/scénario

La cryptographie moderne est principalement basée sur la théorie mathématique et la pratique informatique. Les algorithmes cryptographiques sont conçus en fonction d'hypothèses de complexité de calcul, ce qui les rend difficiles, voire impossibles, à déchiffrer pour un hacker. JCrypTool est un outil logiciel open source indépendant de la plateforme, qui fait partie du projet open source CrypTool. JCrypTool est une plate-forme d'apprentissage en ligne extensible qui présente la cryptographie, la cryptanalyse et la sécurité informatique de manière moderne et facile à utiliser. Ce TP utilisera JCrypTool pour présenter les algorithmes cryptographiques classiques, modernes, symétriques et asymétriques.

Ressources requises

PC avec **CSE-LABVM** installé dans VirtualBox : https://www.netacad.com/resources/lab-downloads?courseLang=en-US

Ou bien télécharger et utiliser JCrypTool directement dans votre PC : https://www.cryptool.org/en/jct/downloads/

Instructions

Partie 1 : Utiliser un algorithme de chiffrement classique

En cryptographie, un chiffrement est un algorithme permettant d'effectuer le chiffrement ou le déchiffrement. Un chiffrement est un ensemble d'étapes (un algorithme) permettant d'effectuer à la fois un chiffrement et le déchiffrement correspondant. Les premiers chiffrements de la cryptographie étaient conçus pour permettre le chiffrement et le déchiffrement à la main, tandis que ceux qui sont développés et utilisés aujourd'hui ne sont possibles que par l'utilisation d'ordinateurs. Les algorithmes classiques sont ceux qui ont été inventés jusque dans les années 50.

Le chiffrement Cesar, également appelé chiffrement à décalage, est l'une des techniques de chiffrement les plus simples et les plus connues. Cette méthode doit son nom à Jules César, qui l'utilisait dans sa correspondance privée. Le code Cesar est un type de chiffrement par substitution dans lequel chaque lettre du texte en clair est remplacée par une lettre située à un certain nombre de positions de l'alphabet. Par exemple, avec un décalage vers la gauche de 3, D serait remplacé par A, E deviendrait B, etc.

Étape 1: Lancez CSE-LABVM.

Étape 2: Ouvrez et explorez JCrypTool.

a. Double-cliquez sur l'icône **jcryptool** sur le bureau. Le répertoire **jcryptool** s'ouvre.

b. Double-cliquez sur l'icône **JCrypTool.**

L'outil comporte quatre fenêtres :

- **Explorateur de fichiers** : permet de localiser, d'ouvrir et d'enregistrer des fichiers.
- **Help (Aide)** : permet de localiser les fichiers d'aide et les tutoriels.
- **Fichier actuellement ouvert :** contient les fichiers qui doivent être traités avec des outils cryptographiques. Le fichier **unsaved001.txt** doit être ouvert.
- **Crypto Explorer :** permet d'accéder aux outils cryptographiques. Par défaut, l'explorateur de chiffrement ne s'affiche pas. Pour l'ouvrir, cliquez sur **Window** > **Show View** > **Crypto Explorer.**

Étape 3: Utilisez l'algorithme Cesar pour chiffrer un message texte.

a. Pour commencer, vous devez renseigner le fichier actuellement ouvert avec le message que vous souhaitez chiffrer. Mettez en surbrillance tout le texte du fichier ***unsaved001.txt** et remplacez-le par le message suivant :

LA CRYPTOGRAPHIE EST AMUSANTE. POUVEZ-VOUS LIRE CE MESSAGE SECRET ?

b. Dans **Crypto Explorer,** cliquez sur **Classic** (Classique) s'il n'est pas développé et double-cliquez sur **Caesar.**

c. Dans la section **Operation** (Opération), sélectionnez **Encrypt** (Chiffrer) si ce n'est pas déjà fait.

d. Dans la section **Alphabet** , vérifiez que les options **Select alphabet** (Sélectionner l'alphabet) et « **Upper Latin (AZ)**) sont sélectionnées. Si ce n'est pas le cas, sélectionnez-les maintenant.

e. Dans la section **Key** (Clé), remplacez la **Enter key using a character** (touche Entrée à l'aide d'un caractère) par **K**. Conservez les valeurs par défaut pour toutes les autres options.

f. Cliquez sur **Finish** (Terminer) pour enregistrer les options et chiffrer les données.

g. Un nouveau fichier nommé ***out001.txt** s'ouvre avec le message chiffré.

Étape 4: Déchiffrez le texte chiffré avec l'algorithme Cesar.

a. Déplacez le fichier ***out001.txt** dans la fenêtre de l'explorateur de fichiers, si nécessaire. Cela garantit que **Crypto Explorer** utilisera ce fichier comme fichier actif. Vous pouvez également fermer le fichier ***unsaved001.txt**.

b. Dans l'onglet **Crypto Explorer**, double-cliquez à nouveau sur l'algorithme **Caesar**.

c. Dans la section **Operation** (Opération) , sélectionnez **Decrypt** (Déchiffrer).

d. Sélectionnez les mêmes paramètres pour déchiffrer le texte chiffré actuel dans le fichier de sortie ***out001.txt**.

e. Cliquez sur **Finish** (Terminer) pour enregistrer les options et déchiffrer les données.

f. Fermez tous les fichiers dans **File Explorer** (l'explorateur de fichiers). Il n'est pas nécessaire de les enregistrer.

Étape 5: Modifiez les paramètres de l'algorithme Cesar.

a. Créez un nouveau fichier texte d'entrée en sélectionnant **File** (Fichier) > **New** (Nouveau) > **Empty File in Texteditor** (Fichier vide dans l'éditeurde texte).

b. Saisissez le message suivant : **Cryptography is fun. Pouvez-vous lire ce message secret ?**

c. Dans l'onglet **Crypto Explorer**, double-cliquez à nouveau sur l'algorithme **Caesar**.

d. **Chiffrer** doit déjà être sélectionné. Pour **Select alphabet** (Sélectionner l'alphabet), définissez la valeur sur la **Upper and lower Latin (A-Z, a-z)** (valeur latin supérieur et inférieur (A-Z, a-z)). Pour la quantité de décalage le long de l'alphabet, définissez la valeur sur **13**.

e. Cliquez sur **Finish** (Terminer) pour enregistrer les options et chiffrer les données.

f. Fermez tous les fichiers dans **File Explorer** (l'explorateur de fichiers). Il n'est pas nécessaire de l'enregistrer.

Exploration plus approfondie

Expérimentez par vous-même avec des algorithmes de cryptographie de type « Cesar » et d'autres algorithmes classiques de cryptographie.

Partie 2 : Utilisez un algorithme de chiffrement symétrique moderne

Dans cette partie, vous allez utiliser un algorithme de chiffrement symétrique moderne. L'une des versions les plus populaires d'un algorithme cryptographique moderne est Advanced Encryption Standard (AES). AES est un chiffrement cryptographique symétrique logiciel et matériel qui est utilisé dans le monde entier pour chiffrer les données sensibles. Le chiffrement AES nécessite une clé de chiffrement pour contrôler le processus de chiffrement et de déchiffrement. Cet algorithme est considéré comme un protocole cryptographique fort en raison de sa complexité et de sa longueur de clé de 128 bits.

Étape 1: Utilisez le chiffrement AES pour chiffrer un message texte.

a. Créez un nouveau fichier texte d'entrée en sélectionnant **File** (Fichier) > **New** (Nouveau) > **Empty File in Texteditor** (Fichier vide dans l'éditeurde texte).

b. Saisissez le message suivant : **Cryptography is fun. Pouvez-vous lire ce message secret ?**

c. Sous l'onglet **Crypto Explorer**, cliquez sur **Symmetric** pour le développer, si nécessaire, puis double-cliquez sur **AES**.

d. Utilisez les paramètres suivants.

- Opération: **Encrypt**
- Source de la clé: **Custom key**
- Longueur de la clé: **128**
- Clé (hex): **AA 00 00 00 00 00 00 00 00 00 00 00 00 00 00 FF**
- Mode: **(ECB) Electronic Codebook**
- Remplissage: **PKCS#5 Padding**

e. Cliquez sur **Terminer**. Un fichier de sortie avec une extension.bin s'ouvre. Vous verrez quatre lignes avec 16 valeurs hexadécimales dans chaque ligne. Le texte chiffré s'affiche à droite de chaque ligne.

Étape 2: Utilisez AES pour déchiffrer un message texte.

a. Double-cliquez à nouveau sur **AES**.

b. Remplacez l'**opération** par **Decrypt**, puis utilisez les paramètres de l'étape 1 pour déchiffrer le texte chiffré.

c. Cliquez sur **Finish** (Terminer) pour enregistrer les options et déchiffrer les données. Un fichier de sortie avec une extension.bin s'ouvre avec le texte déchiffré.

d. Fermez tous les fichiers.

Partie 3 : Utiliser un algorithme de chiffrement asymétrique moderne

Dans cette partie, vous allez utiliser un algorithme de chiffrement asymétrique moderne. Contrairement au chiffrement symétrique, le chiffrement asymétrique chiffre et déchiffre les données à l'aide de deux clés cryptographiques distinctes, mais connectées mathématiquement. Ces clés sont appelées « clé publique » et « clé privée ». Pour qu'une personne envoie un message chiffré à une autre personne à l'aide du chiffrement asymétrique, elle lui demande une clé publique, puis l'utilise pour chiffrer un message avec un algorithme convenu. L'autre personne déchiffre le message à l'aide de sa clé privée. Le message ne peut pas être déchiffré à l'aide de la clé publique.

Étape 1: Utilisez le chiffrement asymétrique RSA pour chiffrer un fichier texte.

a. Créez un nouveau fichier texte d'entrée en sélectionnant **File** (Fichier) > **New** (Nouveau) > **Empty File in Texteditor** (Fichier vide dans l'éditeurde texte).

b. Saisissez le message suivant : **Cryptography is fun. Pouvez-vous lire ce message secret ?**

c. Dans l'explorateur de chiffrement, cliquez sur **Asymmetrical** (Asymétrique) pour le développer et cliquez deux fois sur **RSA** pour ouvrir les paramètres de l'algorithme.

d. Utilisez les paramètres suivants :

- Opération : **Encrypt**
- Magasin de clés : Click **Create a new pair in the keystore**.
- Dans la boîte de dialogue **New key pair** (Nouvelle paire de clés), saisissez les informations suivantes:
  - Nom du contact : **John Smith**
  - Mot de passe : **Secret**
  - Laissez toutes les autres entrées par défaut.

e. Cliquez sur **Terminer**.

f. Cliquez sur **Finish** (Terminer) dans la boîte de dialogue **RSA - encryption** (RSA - Chiffrement) pour chiffrer les données. Un fichier de sortie portant l'extension.bin s'ouvre avec le texte chiffré.

Étape 2: Utilisez le chiffrement asymétrique RSA pour déchiffrer un fichier texte.

a. Double-cliquez sur **RSA**.

b. Sélectionnez **Decrypt** pour l'opération.

c. Select Key = **“John Smith” – public key – 1024**

d. Cliquez sur **Finish** (Terminer) pour déchiffrer le texte chiffré.

e. Saisissez le mot de passe **Secret**. Cliquez sur **OK**.