Lastenheft MATLAB®

Brute-Force-Software mittels verschiedener Verschlüsselungs-algorithmen

Teammitglieder: Adrian Gonzalez
Bruno Hürzeler

Dozent: Prof. Dr. Norbert Hofmann

Fach Matlab-Workshop (matl)

Studiengang: Studiengang ST-Technik
Fachhochschule Nordwestschweiz, Hochschule für Technik

Version	Autor	Datum	Status	Kommentar
0.1	Hub	05.04.2018	provisorisch	Erstellung des Dokuments
1.1	Hub	17.04.2018	provisorisch	Überarbeitung des Dokuments

Inhaltsverzeichnis

1 Zielbestimmung

In dieser Projektarbeit wird eine Software entwickelt, welche für das Entschlüsseln von Passwörtern oder Hashes genutzt wird. Die Entschlüsselung soll mit der *Brute-Force-Methode* realisiert werden. Da in der heutigen Zeit der Digitalisierung der Schutz der persönlichen Daten immer wichtiger wird, kann anhand dieser Software z.B. die Stärke des Passwortes bestimmt werden.

1.1 Muss

1. Evaluierung der vorhandenen Systemressourcen
2. Eingabe / Einlesen von Passwörter -oder Hashes
3. Auswahl des Verschlüsselungsalgorithmus
4. Grafische Benutzer Oberfläche für die Bedienung
5. Ressourcenauswahl für die Berechnungen
6. Ressourcenmonitor
7. Visualisieren der Parameter CPU/GPU-Temperatur und Lüfterdrehzahl
8. Fortschrittsanzeige (ProgressBar)
9. Ausgabe des geknackten Passwortes
10. Logfenster für Informationen

1.2 Wunsch

1. Generieren von CSV-Exports mit allen relevanten Informationen
2. Berechnungen mittels MATLAB Distributed Computing Server (Cloud)
3. Menüleiste mit Raste File (New File, Export CSV, Exit) und Info (About, ..)

1.3 Abgrenzungskriterien

Das Programm wird nur für eigens eingegebene Passwörter oder Hashes verwendet. Für die Verwendung zum Knacken von vertrauenswürdigen Daten oder Dokumente ist diese Software auf keinen Fall geeignet.

Zugelassene Zeichen für die Passwörter sind Zahlen (0-9) und Buchstaben (A-Z / a-z). Sonderzeichen sind nicht Bestandteil. Die Anzahl der Zeichen ist auf x = 8 limitiert. Der Grund für diese Eingrenzung liegt darin, dass das Passwort oder der Hash mit einem Computer in einer sinnvollen Zeit ermittelbar ist.

Mögliche Kombinationen: $10 + 26 + 26 = 62^x = 62^8 = 2.18 * 10^{14}$

2 Produkteinsatz

2.1 Anwendungsbereich

Das Programm findet weder kommerzielle noch projektspezifische Anwendung. Es ist lediglich dazu gedacht, sich etwas in die Thematik Kryptographie einzuarbeiten und die Möglichkeiten der heutigen Verschlüsselungstechnik vor Augen zu führen.

2.2 Zielgruppe

Die Zielgruppe ist der Dozent sowie die Entwickler selbst.

3 Produkteumgebung

3.1 Software

Die Software verwendet je nach Workstation die MATLAB-Toolbox *„Parallel Computing Toolbox“*, welche ab Version R2015a verfügbar ist.

3.2 Betriebssystem

Die Parallel Computing Toolbox ist grundsätzlich auf den folgenden 64-Bit Betriebssystemen ausführbar:

- Windows
- (Wunsch) OS X
- (Wunsch) Linux

Die Software soll für Windows 10 vollständig funktionsfähig sein. Als Wunsch soll die Software zusätzlich auf macOS Sierra funktionieren.

3.3 Hardware

Bevor die Software auf dem Rechner ausgeführt werden kann, müssen die in der Software enthaltenen Clusters validiert werden. Mittels dieser Cluster kann entschieden werden, wie die Rechenoperationen auf die vorhandene Hardware verteilt wird.

CPU Berechnungen

- PC / Laptop mit mindestens 1GB RAM
- Mindestens 5GB freier Festplattenspeicher für temporäre Daten

GPU Berechnungen

- CUDA-fähige NVIDIA GPUs mit Rechenkapazität 3.0 oder höher
- Für die Releases 17b und früher reicht die Rechenfähigkeit 2.0 aus

4 Produktfunktionen

/F10/ Überprüfung der vorhandenen Systemressourcen

/F20/ Einlesen des Passworts oder des Hashs

/F30/ Überprüfung der Länge des eingegebenen Passwortes

/F40/ PC-Hardware (CPU), Grafikkarte (GPU)-Informationen und Temperatur auslesen

/F50/ Dynamisierung des Ressourcenmonitors

/F60/ Beschreiben des Output-Logs

/F70/ Brute-Forcing mit ausgewähltem Algorithmus

/F80/ Schreiben der Daten bei erfolgreichem Ausführen der Software (Passwort, benötigte Zeit, Anzahl Kombinationen

5 Produktdaten

/D10/ Das entschlüsselte Passwort wird auf dem UI ausgegeben.

/D20/ Die CPU- / GPU-Belastung wird als 2D-Plot in Prozent als Funktion der Zeit auf dem UI ausgegeben.

/D30/ Logs mit Angaben zu Versuchsnummer, aktueller Zeit, Belastung der Ressourcen, versuchte Kombinationen werden auf dem UI ausgegeben.

/D30/ {Wunsch}: Die Logdaten sollen als csv-Datei abgespeichert werden können.

6 Produkt-Leistungen

/L10/ Die Software soll jene Passwörter, welche den korrekten Aufbau aus *Kapitel 1.3* aufweisen, knacken können

/L20/ Die graphische Darstellung der Belastung der Ressourcen soll flüssig dargestellt werden können.

/L30/ (Wunsch) Während der Entschlüsselung soll der Fortschritt in einem Balken angezeigt werden.

7 Benutzungsschnittstelle

/B10/ Die Software wird in einem Fenster angezeigt

/B20/ Die Eingabe der Passwörter / Hashs erfolgen durch die Tastatureingabe.

/B30/ Hinweise und Fehlermeldungen werden im Log Fenster ausgegeben. Ausnahme mit der Überprüfung der Passwortlänge, diese wird zur Laufzeit (während der Eingabe) überprüft. Treten dabei Fehler auf, werden diese in einem Prompt visualisiert.

/B40/ Erweiterte Einstellungen sind via Dropdown-Liste und Radio-Buttons mit der Maus anzuwählen

8 Qualitätsbestimmung

Qualitätskriterium	sehr gut	gut	normal	nicht relevant
Änderbarkeit (des Codes)			x	
Überprüfbarkeit (des Codes)			x	
Verständlichkeit (des Codes)			x	
Wartbarkeit (des Codes)			x	
Benutzungsfreundlichkeit (Programm)		x		
Effizienz (Programm & Code)	x			
Funktionale Korrektheit (Programm)		x		
Funktionale Vollständigkeit (Programm)		x		
Robustheit gegenüber dem Benutzer (Programm)			x	

Der Bewertung für den Punkt 'Effizienz' ist insbesondere auf die Algorithmik des eigentlichen Brute-Forcings bezogen.

9 Testfälle

/E10/ Funktionstest von den Bedienelementen (Evaluate System, Start Brute-Forcing, gesamte Menüleiste, Auswahl Ressourcen).

/E20/ Fehlerhafte Zeichen (_, /, +, @, %, ?) für ein Passwort eingeben

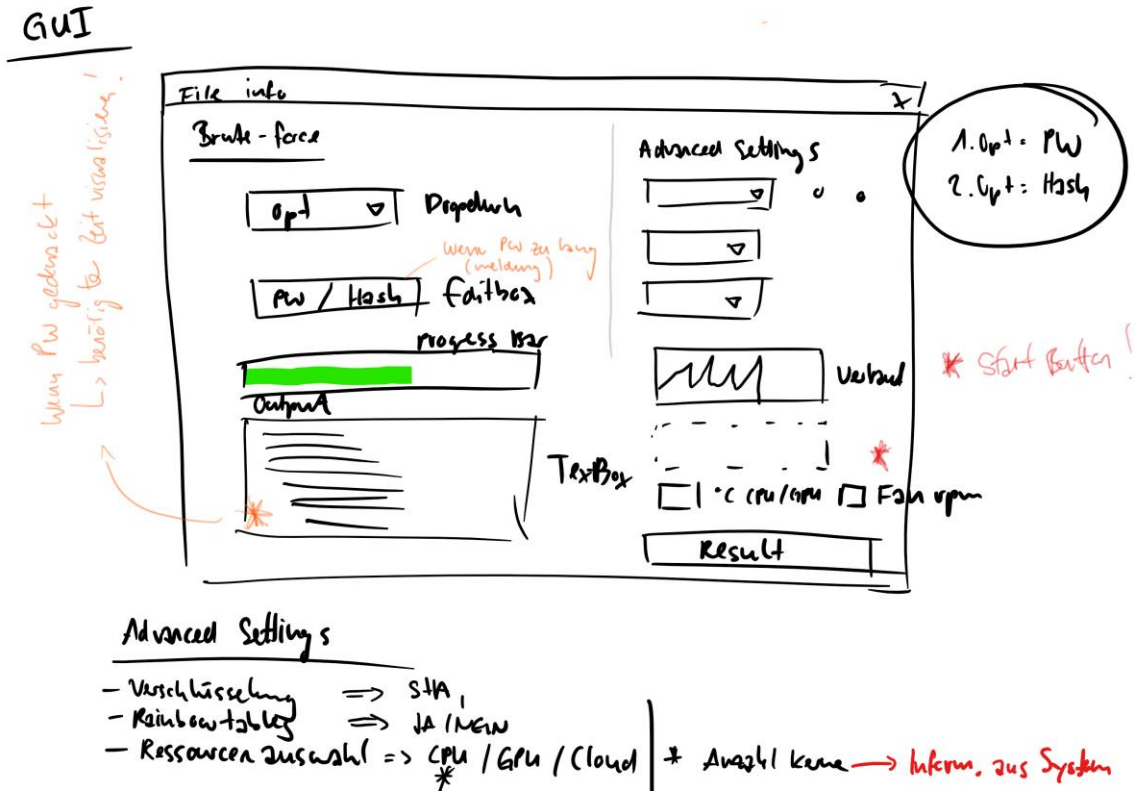
/E30/ Zu viele Zeichen (von 9 Zeichen bis 15 Zeichen) für ein Passwort

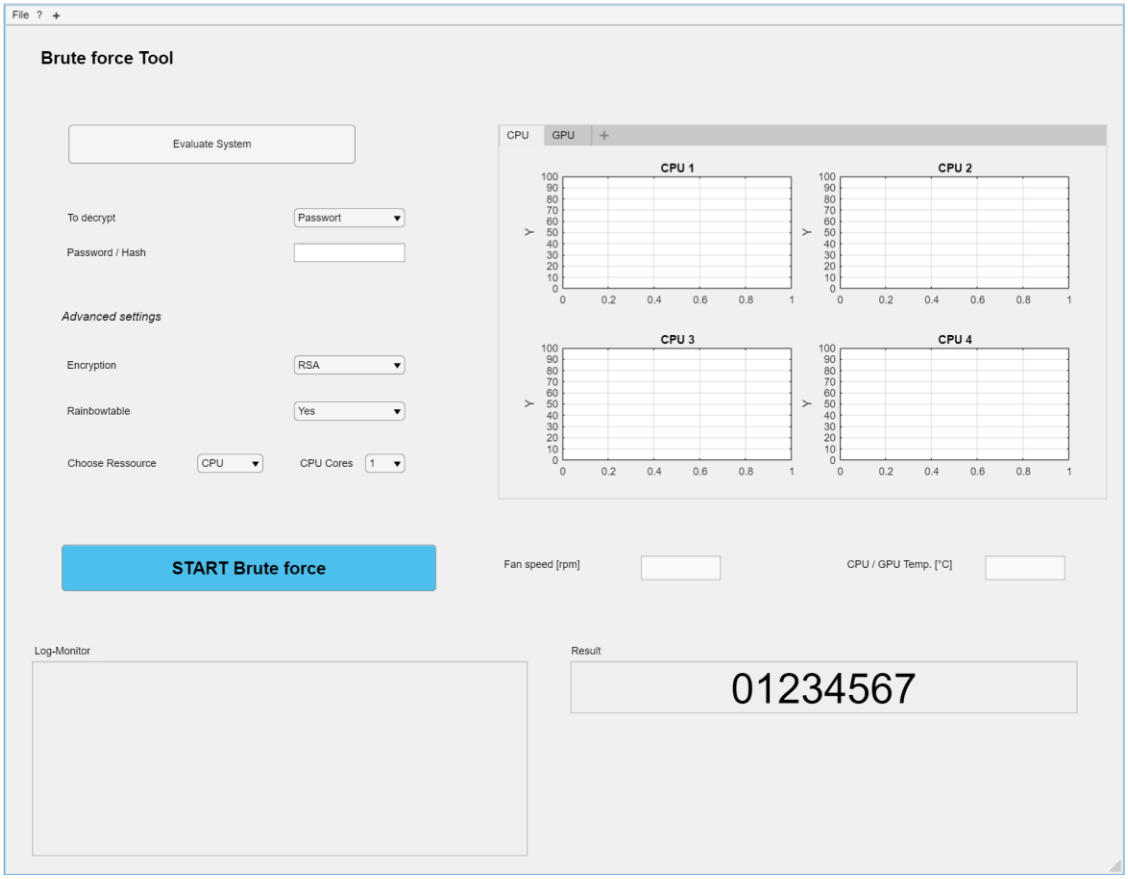
/E40/ Das Programm auf den Rechnern von A.Gonzalez & B. Hürzeler testen.

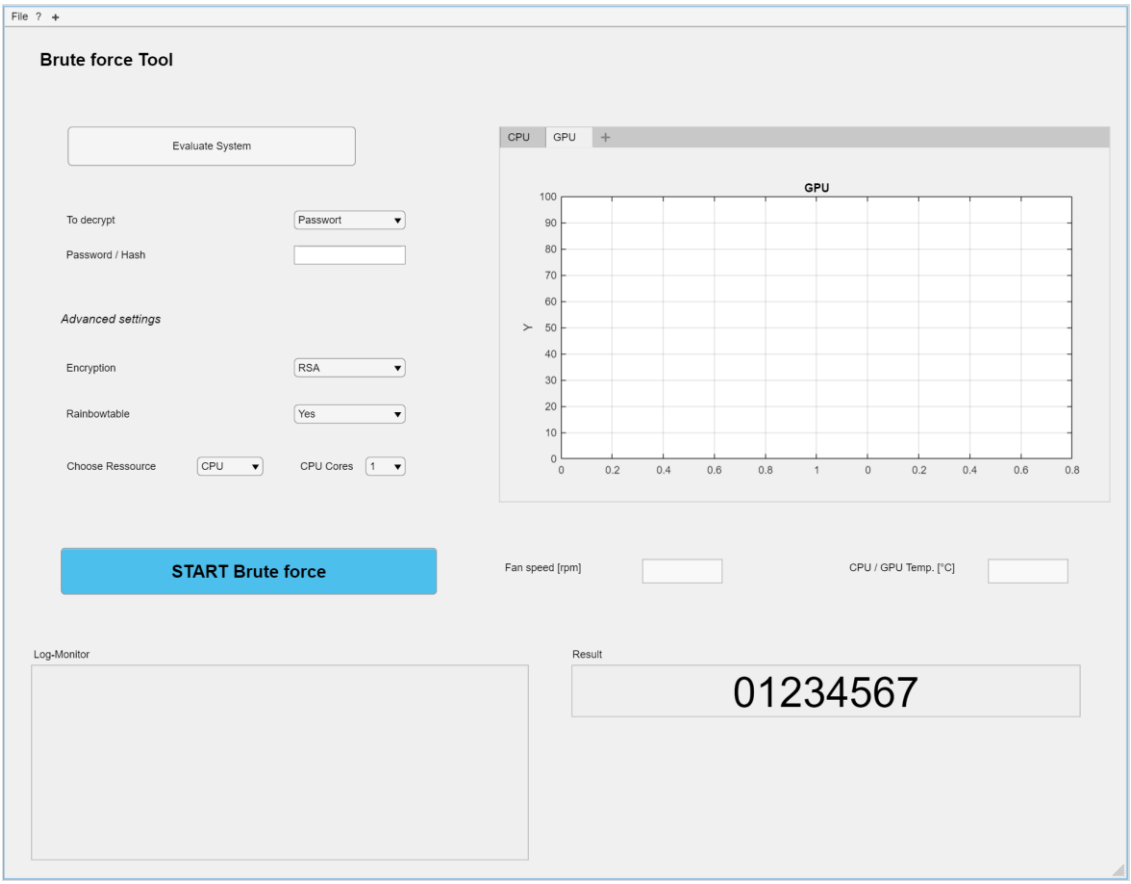
/E50/ Fehlerhandling testen. (GPU-Version nicht valid, MATLAB®-Version nicht valid, Cluster nicht geprüft, Parallel Computing Toolbox nicht installiert, CSV-Export prüfen.

10 Anhang

Entwurf Skizzen User Interface




File ? +
Brute force Tool
Evaluate System
To decrypt
Passwort
Password / Hash
Advanced settings
Encryption
RSA
Rainbowtable
Yes
Choose Ressource
CPU
CPU Cores
1
START Brute force
CPU
GPU
CPU 1
CPU 2
CPU 3
CPU 4
Fan speed [rpm]
CPU / GPU Temp. [°C]
Log-Monitor
Result
01234567


File ? +
Brute force Tool
Evaluate System
To decrypt
Passwort
Password / Hash
Advanced settings
Encryption
RSA
Rainbowtable
Yes
Choose Ressource
CPU
CPU Cores
1
START Brute force
CPU
GPU
GPU
Fan speed [rpm]
CPU / GPU Temp. [°C]
Log-Monitor
Result
01234567