Lehrstuhl Technik der Rechnernetze
Institut für Informatik und Wirtschaftsinformatik
Web: https://www.tdr.wiwi.uni-due.de/

UNIVERSITÄT DUISBURG ESSEN

Network and Information Security

Programmierprojekt

Häufig gestellte Fragen

Einleitung

Dieses Dokument enthält einige häufig gestellte Fragen zum Programmierprojekt und die entsprechenden Antworten darauf. Sollten Sie eine Frage haben, die nicht in diesem Dokument oder der Aufgabenstellung zum Programmierprojekt beantwortet wird, so nutzen Sie bitte das Moodle-Forum zum Programmierprojekt, um Ihre Frage zu stellen.

Häufig gestellte Fragen

1. **Darf ich eine Schleife benutzen um eine ASCII-Tabelle aufzubauen, um nicht sämtliche Zeichen manuell eingeben zu müssen?**

   Ja, eine ASCII-Tabelle darf per Schleife aufgebaut werden.

2. **Die Java-Operatoren % und ^ für Modulo und XOR-Verknüpfung werden in der Aufgabenstellung nicht als verboten deklariert. Dürfen diese Operatoren also verwendet werden?**

   Wir raten Ihnen zu der Verwendung Ihrer eigenen Funktionen, um sicherzustellen, dass diese auch tatsächlich korrekt funktionieren.
   Sie dürfen jedoch auch die Java-Operatoren für die späteren Aufgaben verwenden, falls ihr Code sonst unübersichtlich wird.
   Allerdings müssen Sie dennoch eine eigene Modulo- sowie XOR-Funktion implementieren, in denen Sie natürlich **nicht** die Java-Operatoren verwenden dürfen!

3. **Werden die Blöcke bei den AES-Aufgaben zeilen- oder spaltenweise eingelesen?**

   Alle AES-Blöcke werden spaltenweise ein- und ausgegeben.

4. **Was ist bei Aufgabe 14 (AES: Initiale & zwei weitere Runden) mit „Keyroom“ gemeint?**

   Bitte schauen Sie sich die Folie 18 des Kapitels 6 der Vorlesungsunterlagen oder den Abschnitt 5.2 „Key Expansion“ in der für Kapitel 6 vorgeschlagenen Literatur an.
   Dort heißt es:
   "The Key Expansion generates a total of Nb (Nr + 1) words" und "The resulting key schedule consists of a linear array of 4-byte words, denoted [wi], with i in the range 0 <= i < Nb(Nr + 1)."
   Der in der Aufgabenstellung angegebene Keyroom gibt dabei die maximale Länge des "key schedulings" in Byte an. Also 4 * Nb * (Nr + 1).
   Da der Server momentan nur Aufgaben für AES-128 generiert, ist es möglich, dass Sie ohne diese Angabe auskommen, da Sie in diesem Fall immer konstant 176 beträgt.

5. **Bei den letzten Aufgaben (asymmetrische Verschlüsselung) sind die Ergebnisse teilweise deutlich größer als der Wertebereich von int bzw. long (z.B. bei Diffie-Hellman oder RSA, wenn man große Zahlen potenziert). Dürfen wir in solchen Fällen auf komplexe Datentypen von Java wie z.B. BigInteger zurückgreifen?**

   BigInteger kann prinzipiell genutzt werden. Die Einschränkungen bezgl. Java-eigener Routinen gelten aber weiterhin. D. h. Modulo, etc. muss selbst implementiert werden, was auch zur Folge hat, dass insbesondere BigInteger.modPow(...) nicht benutzt werden darf.

   Zu Übungszwecken (!) ist es aber vollkommen ausreichend eine Implementierung zu erstellen, die mit so kleinen Primzahlen arbeitet, dass die Wertebereiche von int oder long nicht überschritten werden.

6. **In welchem Format muss der Quelltext abgegeben werden?**

   Sollten Sie nur die Datei Client.java modifiziert haben, laden Sie bitte nur diese per TMT hoch. Andernfalls laden Sie bitte ein ZIP-Archiv hoch, welches alle veränderten und neuen .java-Dateien enthält. Beachten Sie außerdem die weiteren Hinweise zur Abgabe in der Aufgabenstellung des Programmierprojektes

7. **Kann der Quelltext nach dem Hochladen noch verändert werden?**

   Ja. Sie können Ihren Quelltext beliebig oft hochladen. Es wird nur die zuletzt hochgeladene Version bewertet. Beachten Sie außerdem die weiteren Hinweise zur Abgabe in der Aufgabenstellung des Programmierprojektes