**Асимметричные криптосистемы.**

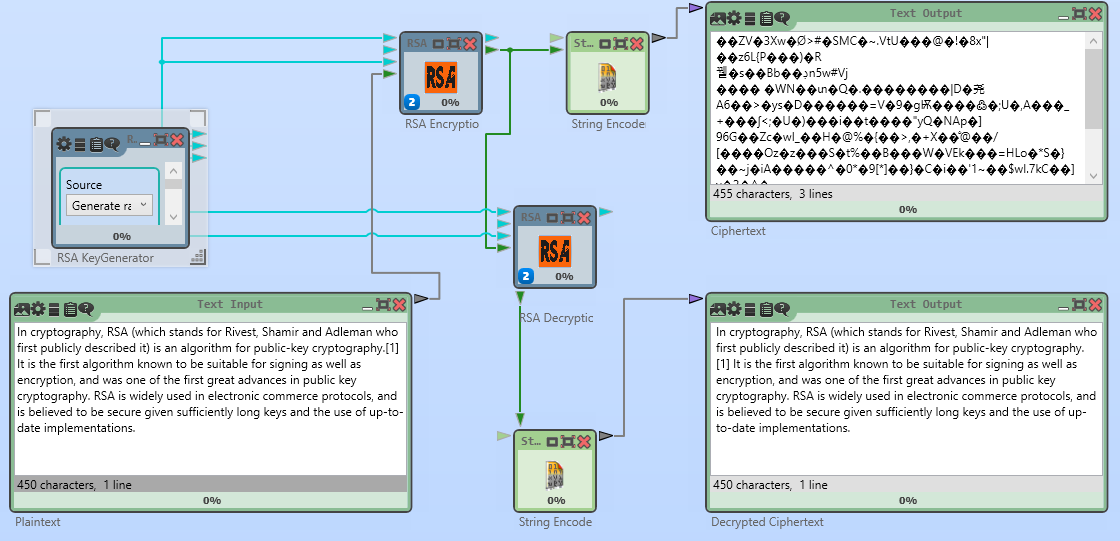
**Цель:** изучить основные принципы работы асимметричных криптосистем на примере алгоритма RSA.

**Задачи практической работы**

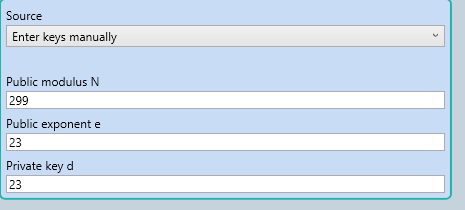
1. Проанализировать эмуляцию алгоритма RSA и примитивных атак на шифр, используя Cryptool 2. Выделить основные необходимые настройки шифра и требуемые ограничения на параметры.
2. Программно реализовать и модифицировать любую асимметричную криптосистему. В случае отсутствия опыта программирования подойдет реализация алгоритма на псевдокоде или в виде блок схем, включающих основные этапы алгоритма с отображением формул и основных математических действий. Атаки и модификации, приведенные ниже, указаны для RSA. Если атака или модификация не применима для реализуемого алгоритма разрешается найти любую альтернативу (атаки, применимой к алгоритму; модификации для ускорения алгоритма и дополнительной защиты).
3. Для созданной реализации криптосистемы предлагается провести примитивный криптоанализ на устойчивость к следующим атакам, а также сделать минимальные модификации по оптимизации (ускорению процессов шифрования, дешифрования, процесса генерации ключей).

**Порядок выполнения работы.**

* + - 1. Для визуализации алгоритма RSA предлагается использовать следующий шаблон Cryptool 2 (Templates -> cryptography -> modern -> asymmetric-> RSA Cipher). Укажите в данном шаблоне свои входные данные и криптографический ключ



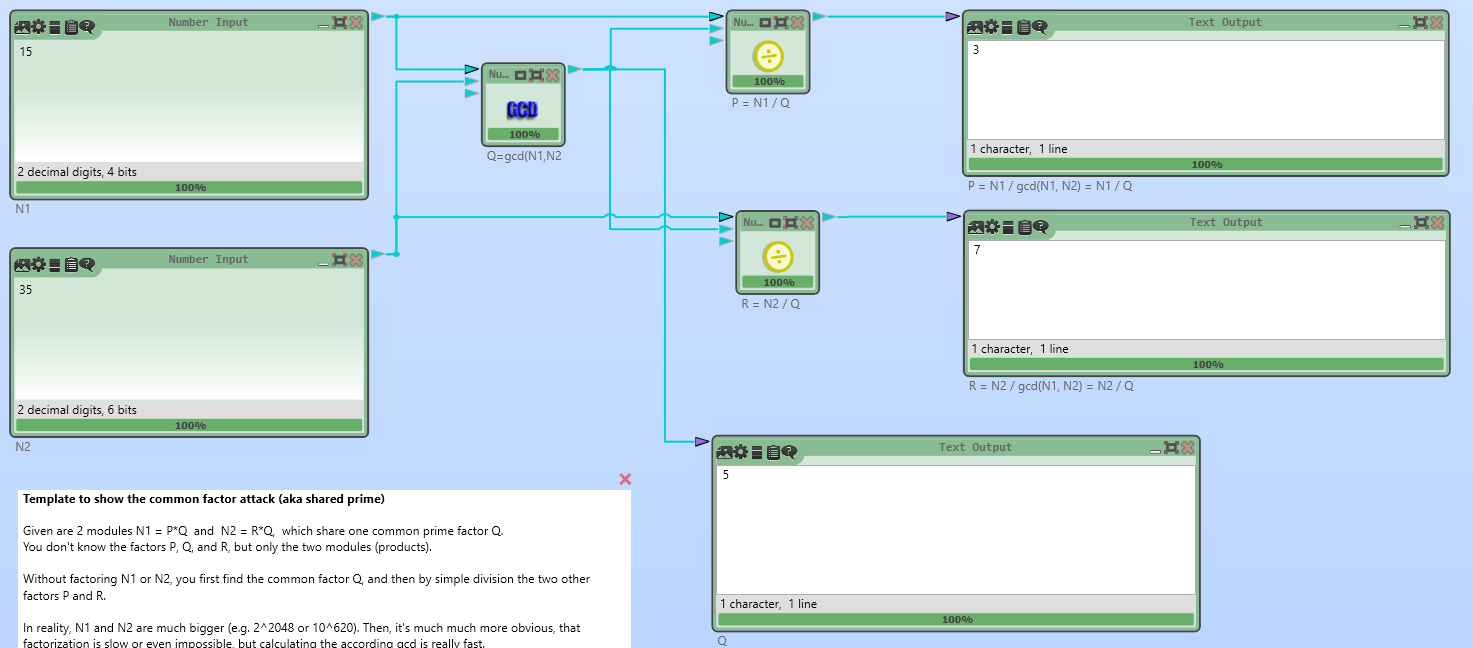
Проанализируйте каждый из блоков шаблона. Задайте параметры ключевой пары RSA вручную для этого откройте настройки блока RSA KeyGenerator и задайте собственные значения параметров N, e, d



Обратите внимание, что данные параметры не являются случайными и задаются по определенным правилам алгоритма RSA. В частности, открытый модуль N является произведением двух простых чисел. Шифрующая экспонента e является мультипликативным обратным к дешифрующей экспоненте d по модулю функции Эйлера от N. Изучите дополнительные ограничения на данные параметры (какой они должны быть длины, являются ли они простыми числами, как соотносятся с другими параметрами алгоритма RSA).

Для блоков RSA Encryption и RSA Decryption укажите как выглядят формулы для шифрования и дешифрования с учетом созданных ранее ключей.

* + - 1. Выполните атаку на алгоритм RSA на основе общего делителя для модуля шифрования. Выделите основные этапы проведения атаки, при каких условиях атака возможна.



Зашифруйте два шифротекста с помощью различных модулей шифрования N c общим делителем. Покажите, как эти шифротексты могут быть дешифрованы при неизвестных дешифрующих экспонентах d.

1. Создать программную реализацию асимметричной криптосистемы. Ниже будут даны атаки и модификации, предлагаемые для реализации криптосистемы RSA, однако могут быть разработаны любые другие асимметричные криптосистемы (например, криптосистемы Эль-Гамаль, Рабина, Пейэ и пр.). Язык программирования выбирается на усмотрение студента.
2. Модифицировать программную реализацию, созданную в пункте 5, чтобы повысить эффективность процедуры шифрования/дешифрования, генерации ключей, устойчивость к определенным атакам. Предлагается выполнить хотя бы 3 из нижепредставленных модификаций:
   * использование китайской теоремы об остатках для ускорения процесса дешифрования RSA;
   * использование алгоритма быстрого возведения в степень для ускорения процесса шифрования RSA;
   * показать, как используются алгоритмы дополнений (padding) в асимметричных криптосистемах, например, из стандарта PKCS#7;
   * добавить в программную реализацию RSA ограничения на параметры, дополнительные механизмы, тесты и проверки, которые позволят защититься от:
     + атак «человек посередине»;
     + атака хастада или любой другой атаки на малую экспоненту;
     + атака по времени выполнения на несбалансированные ветки алгоритма;
     + атаки на RSA при неудачном выборе параметров криптосистемы (модуль N является произведением простого числа Марсенна 8191 и простого числа Ферма 65537, простые числа p и q слишком близки друг к другу и пр.);
     + атаки повторным шифрованием;
     + атаки методом ферма.

Лабораторная работа предусматривает программную реализацию. В качестве **альтернативного варианта** можно рассмотреть любую другую асимметричную криптосистему. Требуются именно криптосистемы, которые содержат процессы генерации ключевой пары, шифрование и дешифрования, алгоритмы согласования ключей, например алгоритм Диффи-Хэллмана, не подходят для реализации. Для лабораторной работы достаточно реализовать одну асимметричную криптосистему в режиме шифрования или в режиме подписи.

Основные процессы, которые необходимо реализовать в самом шифре:

* процесс шифрования;
* процесс дешифрования;
* процесс генерации ключевой пары (генерация открытого и закрытого ключа).

Если реализуется подпись, то основные процессы, которые необходимо реализовать в самом алгоритме:

* процесс генерации ключевой пары;
* процесс подписи;
* процесс проверки подписи

Примеры асимметричных шифров, которые могут быть выбраны для реализации:

* RSA (криптосистема RSA должна быть модифицирована согласно пункту 6 лабораторной, т.е. нужно реализовать не менее 3 модификаций)
* Эль-Гамаль
* Рабина

Также асимметричная криптосистема может быть реализована в виде цифровой подписи, в частности могут быть выбраны следующие алгоритмы для реализации:

* подпись RSA (подпись RSA должна быть модифицирована согласно пункту 6 лабораторной, т.е. нужно реализовать не менее 3 модификаций подписи)
* подпись Эль-Гамаль
* ГОСТ 34.10
* DSA/ECDSA

**Вопросы к защите**

1. Проблема распределения криптографических ключей
2. Цели протоколов (проверка активности, аутентификация сторон, распределения ключа, подтверждение ключа, проверка новизны ключа). С помощью каких механизмов достигаются данные цели.
3. Атаки на криптографические протоколы
4. Примеры протоколов аутентификации, распределения/согласования ключей. Точное воспроизведение с указанием всех параметров не требуется, но необходимо уметь воспроизводить и выделять - механизмы обеспечения новизны ключа, правило согласования ключей, механизм транспортировки ключа, механизмы аутентификации сторон.
5. Примеры односторонних функций. Сводимость проблем асимметричной криптографии
6. Алгоритм согласования ключей Диффи-Хеллмана
7. Криптосистемы RSA, Elgamal
8. Эллиптические кривые в криптографии. ECDH