**Протокол**

Два легальных пользователя - Алиса и Боб, обладающие общим секретным мастер-ключом **mk**. Перед передачей данных Алиса и Боб проходят процедуру согласования. А именно, они открыто обмениваются случайными числами, защищенными имитовставками и , выработанными с помощью мастер-ключа:

Алиса и Боб проверяют полученные имитовставки. Если они соответствуют случайным числам, то Алиса и Боб на основе мастер-ключа и случайных чисел , вырабатывают сеансовый ключ и начинают передачу зашифрованных данных.

**Ответ / Обсуждение**

Очевидно, что атакующий не сможет сфабриковать случайное число, так как для этого ему нужно создать корректную имитовставку. Однако, он легко может выдать себя за Алису или Боба. Для этого ему достаточно использовать пару открытого числа и соответствующую имитовставку, переданную на предыдущих сеансах (*атака повтора*). Более того, атакующий может выдать себя за Боба даже при первом согласовании. Для этого атакующий просто отвечает Алисе её же парой открытого текста и имитовставки (*атака отражения*).

Чем могут навредить данные атаки?

Судя по протоколу данные атаки не дают возможность узнать мастер-ключ и выработанный сеансовый ключ. Таким образом, конфиденциальность данных не будет нарушена. Тем не менее, можно создать новый, нелегальный сеанс связи, что может негативно сказаться на работе системы. Например, в это время легальный пользователь не сможет согласоваться или атакующий начнет постоянно слать случайные строки – легальная стороны начнет их расшифровывать и получать мусор на входе, что может случайно совпасть с какой-то легальной командой или просто перегрузить систему (*DDoS-атака*).

Что можно сделать, чтобы защититься от подобных атак?

Для защиты от повторов достаточно запоминать случайные числа, переданные на предыдущих согласованиях. Также, можно просто добавить метку времени (также защищенную имитовставкой). Запоминание случайных чисел, переданных на предыдущих согласованиях, также защитит от атаки отражения, если запоминать только что отосланное число. Другой способ – добавить идентификатор отправителя (и защитить имитовставкой). Тогда при получении сообщения всегда первым проверяется идентификатор, если он совпадает с собственным, то сообщение отбрасывается.