|  |  |
| --- | --- |
|  | |
| МИНОБРНАУКИ РОССИИ | |
| Федеральное государственное бюджетное образовательное учреждениевысшего профессионального образования"Российский технологический университет"МИРЭА | |
| Институт информационных технологий (ИТ) |  |
| Кафедра ВТ |  |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №3** | |
| **по дисциплине** | |
| **«Защита информации»** | |
|  | |
| Выполнил студент группы ИВБО-01-15 | Пушкарев Е.В. |
| Принял | Воронков С. О. |

|  |  |  |
| --- | --- | --- |
| Лабораторные работы выполнены | «\_\_»\_\_\_\_\_\_\_201\_\_ г. |  |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_201\_\_ г. |  |

Москва 2018

**Задание:**

Разработать клиент-серверную программу, осуществляющую работу протокола SRP6.

**Решение:**

При работе алгоритма:

N – простое, такое, что N = 2q + 1, q – другое простое число (N = 1097)

g – генератор по mod N: для любого 0 < X < N существует единственный Х, такой что g^X mod N = X (g = 11)

k – параметр, который равен 3.

1. Клиент генерирует случайную строку s (соль)
2. Далее клиент вычисляет хеш-функцию X=H(s, P), где Р – пароль, введенный пользователем
3. Клиент вычисляет значение V = g^X mod N и отсылает серверу I, s, V (I – username, s – salt, V – верификатор).
4. Клиент вычисляет а (случ. число), А = g^a mod N и отправляет серверу I, A
5. Сервер проверяет, что А != 0
6. Сервер вычисляет b (случ. число), B = (k \* v + g ^ b mod N) mod N и отсылает клиенту s и В.
7. Клиент проверяет, что В != 0.
8. Клиент и сервер вычисляют U = H (A, B). Если U = 0, то соединение прерывается.
9. Клиент вычисляет общий ключ сессии (К):

X = H (s, P). S = ((B – k \* (g ^ X mod N)) ^ (a + U \* X)) mod N. K = H(S)

1. Сервер вычисляет ключ сессии (К):

S = ((A \* (V ^ U mod N)) ^ b) mod N. K = H(S)

1. Клиент вычисляет М = H ( H(N) XOR H(g), H(I), S, A, B, K) и отсылает серверу значение М.
2. Сервер проводит ту же самую операцию, что и клиент в пункте 11.
3. Если Мсервер = Мклиент, то успех и сервер отправляет клиенту R = H(A, M, K)
4. Клиент вычисляет свою R и если Rc=Rk, то клиент и сервер те, за кого себя выдают

**Вывод:**

После выполнения описанного алгоритма клиент и сервер получили равные значения R.

**Ссылка на git-репозиторий:**

https://github.com/Kefir103/CryptLabSRP