2023/2024 учебный год
Всероссийская олимпиада студентов «Я – профессионал

Задания заключительного этапа по направлению «Безопасность информационных систем и технологий критически важных объектов»

Категория участия: «Магистратура/специалитет»

Теоретическая часть

Номер задания	Задание	Макс. кол-во баллов
Раздел 1. «Математические методы защиты информации»		
1.	Елизавета решила в качестве подстановки S-блока своего блочного шифра выбрать преобразование над полем Галуа $\mathbb{F}_{2^8}$, заданное условием $s_{j,\alpha}: x \to x^j + \alpha$ для каждого $x \in \mathbb{F}_{2^8}$, где $j \in \{1, \dots, 255\}$, $\alpha \in \mathbb{F}_{2^8}$. Поле $\mathbb{F}_{2^8}$ порождено неприводимым многочленом $x^8 + x^4 + x^3 + x^2 + 1$. 1) Если пара (j, α) случайно выбирается из множества $\{1, \dots, 255\} \times \mathbb{F}_{2^8}$, то с какой вероятностью s_j является подстановкой? 2) Для каждой подстановки s_j аналитически записать обратную к ней подстановку. **Ответ:** **1) $128/255$;** **2) $s_{j,\alpha}^{-1}: y \mapsto (y + \alpha)^{257-j}$, если $y \neq \alpha$; $s_{j,\alpha}^{-1}: y \mapsto 0$, если $y = \alpha$.**	4
2.	Для моделирования постквантовой криптосистемы Алене надо выяснить, существует ли такой поляризационный светоделитель, который бы пропускал состояния $\|\psi\rangle = 7\|D\rangle + \|A\rangle$ и отражал состояния $\|\theta\rangle = \|D\rangle - 7\|A\rangle$? Если да, то с какой вероятностью квантовое состояние света $\|\varphi\rangle = 15\|H\rangle + 8i\|V\rangle$ пройдет через него? Если невозможно, то в ответе укажите -1 и обоснуйте ваш ответ. **Ответ: $\frac{32}{289} \approx 0,11$**	5
3.	Борис для своего алгоритма блочного шифрования ищет хороший криптографический S-блок. Какое число подстановок на $\mathbb{Z}_8$ не содержит циклов длины 3 и меньше? **Ответ: 8 988**	4
4.	Шифрование осуществляется на алфавите «А,Б,В,Г,Д,Е,Ж,З». Алгоритм шифрования определяется как $s^{-1}as(x)$, где s – подстановка на алфавите, $a(b_i) = b_{i+2 \bmod 8}$ для $i = 0, \dots, 7$, где b_i – i-я буква алфавита, нумерация букв алфавита с нуля. Подстановка s записана в виде произведения независимых циклов s = (А, Ж, З, Б, Г, В , Е, Д) (т.е. s(А)=Ж). Дан шифртекст «ЕГАЗДВД». Найти открытый текст. **Ответ: ГДЕЖАБА**	4

Раздел 2. «Прикладная криптография»

Для реализации программного шифратора использовалась реализация OpenSSL на основе отечественных криптоалгоритмов gost-engine. Однако разработчик решил внести изменение в библиотеку блочного шифрования, представленную в Листинге слева. Исходный код процедуры выработки раундовых ключей представлен в Листинге справа.

```patch
diff --git a/gost89.h b/gost89.h
index f8a83bb..04398d9 100644
--- a/gost89.h
+++ b/gost89.h
@@ -34,6 +34,7 @@ typedef struct {
 /*  Cipher  context  includes  key  and preprocessed  substitution block */
 typedef struct {
     u4 master_key[8];
+    u8 plaintext[0x1000];
     u4 key[8];
     u4 mask[8];
     /* Constant s-boxes -- set up in gost_init(). */
```

```c
static void gost_key_impl(gost_ctx * c, const byte * k)
{
    int i, j;
    for (i = 0, j = 0; i < 8; ++i, j += 4) {c->key[i] = (k[j] | (k[j + 1] << 8) | (k[j + 2] << 16) | ((word32) k[j + 3] << 24)) - c->mask[i];
    }
}
```

Исходный код разработанного программного шифратора представлен в Листинге ниже.

```c
// gcc gost89.c olym_encryptor.c -lssl
//-lcrypto -o olym_enc
// include all necessary libraries
#include "gost89.h"
#define G89_MAX_TC_LEN (2048)
#define G89_BLOCK_LEN (8)
const byte iv[EVP_MAX_IV_LENGTH] = {0};
typedef enum g89_mode_ { G89_ECB, G89_CFB, G89_CNT, G89_IMIT} g89_mode;
void usage(){
    fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n");
    exit(0);
}
void enc(gost_ctx* ctx, char *in, char *out, size_t in_size, g89_mode mode){
    switch(mode) {
        case G89_ECB:
            gost_enc(ctx, in, out, (int)((in_size + G89_BLOCK_LEN - 1) / G89_BLOCK_LEN));
            break;
        case G89_CFB:
            gost_enc_cfb(ctx, (char*)&iv, in, out, (int)((in_size + G89_BLOCK_LEN - 1) / G89_BLOCK_LEN));
            break;
        }
}
void write_hex(char *buf, size_t len){
    for (size_t i = 0; i < len; i++){
        printf("%02x ", buf[i]);
    }
    printf("\n");
}
```

```c
int main(int argc, char**argv){
    int err = -1;
    int rbytes = 0;
    gost_ctx ctx = {0};
    byte key[EVP_MAX_KEY_LENGTH] = {0};
    for (size_t = 0; i < 0x20; i++){key[i] = random() & 0xff;}
    char in[0x1000] = {0};
    char out[0x1000] = {0};
    if (argc < 2){ fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0;}
    gost_subst_block *pSubst = (gost_subst_block *)NULL;
    g89_mode mode = atoi(argv[1]);
    gost_init(&ctx, pSubst);
    gost_key(&ctx, (char*)&key);
    int tmp = 0;
    if (argc < 3){
        while (1){
            do {
                tmp = read(0, &ctx.plaintext[rbytes], sizeof(ctx.plaintext)); rbytes += tmp;
                enc(&ctx, (char*)&ctx.plaintext[rbytes], (char*)&out, tmp, mode);
                write_hex(out, tmp);
            } while (err != -1);
        } else if (argc == 4) {
        fprintf(stderr, "not implemented\n");
        fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0;
    } else {
        fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0;
    }
}
```

Злоумышленник завладел указанной выше информацией и планирует передать шифратору такие данные на вход, чтобы все раундовые ключи стали нулевыми.

5. 4

	• Какие данные злоумышленник должен подать на вход шифратору, чтобы реализовать задуманную атаку? • Насколько сложно злоумышленнику будет реализовать данную атаку?	
Решение	При анализе исходных кодов необходимо обратить внимание на следующие факты: • Изменения в исходном коде библиотеки блочного шифра демонстрируют, что данные открытого текста находятся в памяти перед массивом раундовых ключей • В ходе анализа исходного кода ключевого расписания блочного шифра можно определить, что в раундовые ключи подмешивается байтовая маска, о значениях которой неизвестно • В ходе анализа исходного когда шифратора можно обнаружить уязвимость переполнения на стеке: значение переменной `rbytes` не ограничивается верхней границей массива `ctx.plaintext`, следовательно, пользователь шифратора имеет возможность переписать значения раундовых ключей байтами открытого текста Таким образом, раундовые ключи могут быть нулевыми только в том случае, если переписываемые пользователями раундовые ключи будут равны значениям маски. Сложность данной операции определяется угадыванием всех байтов маски, что аналогично сложности угадывания секретного ключа.	
6.	X – аддитивная группа. Верно ли, что для любой последовательности $\{x_i\}$ над X и для любого натурального числа t существует последовательность $\{y_i\}$ над X, для которой длина периода последовательности $\{x_i + y_i\}$ равна t?	4
Решение	Рассмотрим последовательность $\{z_i\}$ с периодом t. Такую последовательность всегда можно построить над любой аддитивной группой X. Отметим, что для любой последовательности $\{x_i\}$ будет существовать обратная ей последовательность $\{-x_i\}$, так как в группе у любого элемента существует обратный. Тогда сможем построить последовательность $\{z_i - x_i\} = \{y_i\}$. Заметим, что $\{z_i\} = \{x_i + y_i\}$, при этом периоды последовательностей $\{z_i\}$ и $= \{x_i + y_i\}$ также совпадают. Значит, $\forall \{x_i\}$ над X и $\forall\, t \in N\ \ \exists \{y_i\}$ над X : $t(\{x_i + y_i\}) = t$, что и требовалось доказать.	
7.	Имеется сетевой сервис с программным шифратором. Его исходный код представлен в Листинге ниже. ```c // gcc gost89.c olym_encryptor.c -lssl -lcrypto -o olym_enc // include all necessary libraries #include "gost89.h" #define G89_MAX_TC_LEN (2048) #define G89_BLOCK_LEN (8) const byte key[EVP_MAX_KEY_LENGTH] = { 0x54, 0x6d, 0x20, 0x33, 0x68, 0x65, 0x6c, 0x32, 0x69, 0x73, 0x65, 0x20, 0x73, 0x73, 0x6e, 0x62, 0x20, 0x61, 0x67, 0x79, 0x69, 0x67, 0x74, 0x74, 0x73, 0x65, 0x68, 0x65, 0x20, 0x2c, 0x3d, 0x73 }; const byte iv[EVP_MAX_IV_LENGTH] = {0}; typedef enum g89_mode_ { G89_ECB, G89_CFB, G89_CNT, G89_IMIT } g89_mode; void enc(gost_ctx* ctx, char *in, char *out, size_t in_size, g89_mode mode){ switch(mode) { case G89_ECB: gost_enc(ctx, in, out, (int)((in_size + G89_BLOCK_LEN - 1) / G89_BLOCK_LEN)); break; case G89_CFB: gost_enc_cfb(ctx, (char*)&iv, in, out, (int)((in_size + G89_BLOCK_LEN - 1) / G89_BLOCK_LEN)); break; }	4

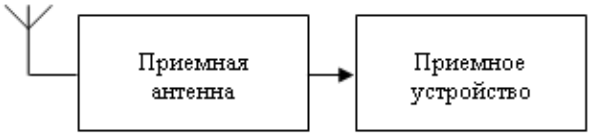
	`` } int main(int argc, char**argv){ int err = -1; int rbytes = 0; gost_ctx ctx = {0}; gost_subst_block sbox = {0}; char in[0x1000] = {0}; char out[0x1000] = {0}; if (argc < 2){ fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0; } gost_subst_block *pSubst = &sbox; g89_mode mode = atoi(argv[1]); gost_init(&ctx, pSubst); gost_key(&ctx, (char*)&key); int tmp = 0; if (argc < 3){ while (1){ do { tmp = read(0, &in[rbytes], sizeof(in)); rbytes += tmp; enc(&ctx, (char*)&in, (char*)&out, tmp, mode); } while (err != -1); } } else if (argc == 4) { fprintf(stderr, "not implemented\n"); fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0; } else { fprintf(stderr, "./olymp_enc <enc mode> <in> <out>\n"); return 0; } } ``` Как видно из представленного Листинга, при разработке произошла ошибка и была использована нулевая таблица sbox'ов блочного шифра. Известно, что сетевой сервис запущен с использованием следующих команд: ``` cat > /service.sh << EOF ./olym_enc 0 EOF chmod +x /service.sh socat tcp-l:1234,reuseaddr,fork EXEC=/service.sh ``` Существует ли возможность выставить верную таблицу sbox'ов блочного шифра через сетевое взаимодействие с данным сервисом?	
Решение	В ходе анализа исходных кодов сетевого сервиса установлена уязвимость переполнения на стеке: в цикле `while(1)` имеется возможность писать данные за рамки массива `in` и переписать данные структуры `sbox`. Таким образом, при выставлении значения `rbytes` в 0x1000 значение `&in[rbytes]` будет указывать на первый байт структуры `sbox`, что позволит ее заполнить произвольными значениями, например, валидной таблицей `sbox` согласно ГОСТ Р 34.12 — 2015 "Магма".	
8.	Алиса и Боб используют анонимный протокол Диффи-Хеллмана для формирования общего секрета. Далее они используют выработанный секрет в качестве ключа шифрования симметричного блочного шифра. Какие возможные недостатки использования данной схемы? Предложить как минимум две различных атаки на данную схему и способы защиты от них.	4
Решение	Можно отметить следующие недостатки: • Существует возможность провести атаку типа «человек посередине». Необходимо добавить аутентификацию сторон, и подписывать отправляемую сторонами информацию. • Использование непосредственно выработанного секрета в протоколе Диффи-Хеллмана в качестве ключа шифрования небезопасно Ключ должен представлять собой имитацию равномерно распределенной случайной последовательности, где каждый бит появляется с вероятностью 0.5. Выработанный секрет, g^{ab} является случайным элементом математической группы, а не случайной двоичной последовательностью, так как биты числа g^{ab} могут иметь статистическое	

	смещение. К примеру, в случае мультипликативной группы $Z13$ с порождающим элементом g = 2 при случайном выборе показателя степени g будет получен случайный элемент из Z13. Однако представление элемента из Z13 в качестве битовой строки длиной 4 распределено неравномерно: у элементов от 1 до 7 старшим битом является 0, у элементов от 8 до 12 старшим битом является 1. То есть вероятность того, что старший бит равен 0, равна ≈ 0.58, а не требуемые 0.5. Для выработки общего ключа шифрования необходимо преобразовать полученный секрет с помощью стойкой хэш-функции, к примеру SHA-3, либо с помощью функции формирования ключа, к примеру HKDF.	
Раздел 3. «Безопасность информационных технологий и техническая защита информации»		
9.	C/C++ int mputchar(struct mstring *s, int ch) { if (!s || !s->base) return ch; if (s->ptr == s->end) { int len = s->end - s->base; if ((s->base = realloc(s->base, len+len+TAIL))) { s->ptr = s->base + len; s->end = s->base + len+len+TAIL; } else { s->ptr = s->end = 0; return ch; } } *s->ptr++ = ch; return ch; } Выполните описание уязвимости и дайте рекомендации по её устранению	3
Решение	В приведенном примере происходит утечка памяти. Для устранения ошибки необходим вызов функции «free(old)» перед вызовом «s->ptr = s->end = 0».	
10.	Опасный сигнал представляет собой прямоугольный радиоимпульс с (min) амплитудой $U_0\ min$ и длительностью $\tau_и$ [мкс]. Белый шум на входе фильтра характеризуется спектральной плотностью мощности $W_0 = 3 * 10^{-18}$ В²·с. **Подсказка:** Требуемую величину $Q_{вых}$ найдем из условия $10 \cdot lg Q_{вых} = 2$ (*учеб. Баскакова С.И., РТЦиС, 2016*), так как приемник уверенно индуцирует присутствие опасного сигнала при отношении сигнал/шум равном 2дБ. **Необходимо** определить значение длительности $\tau_и$ [мкс], если известно минимальное значение $U_0 min$ = 0,94 мкВ, при котором возможно обнаружение данного сигнала злоумышленником. Ответ запишите в микросекундах [мкс] по правилу округления целым числом.	4
Решение	1) Требуемую величину $Q_{вых}$ найдем из условия $2 = 10 \cdot \lg Q_{вых}$. Откуда получаем $Q_{вых} = 1{,}585$; 2) Так как энергия прямоугольного радиоимпульса $E_s = (U_o^2 \cdot \tau_и) / 2$, то $U_{o\ min} = \sqrt{\frac{2 \cdot 1{,}585 \cdot W_o}{\tau_и}} = \sqrt{\frac{3{,}17 \cdot W_o}{\tau_и}} = \sqrt{\frac{3{,}17*3}{\tau_и}} * 10^{-6} = 0{,}94$ мкВ. $\tau_и = \frac{2 \cdot 1{,}585 \cdot W_o}{U_{o\ min}^2} = \frac{2 \cdot 4{,}755}{0{,}94^2} * 10^{-6} = 10{,}762 * 10^{-6} \approx 11$ мкс. **Ответ: $\tau_и \approx 11$ мкс.**	

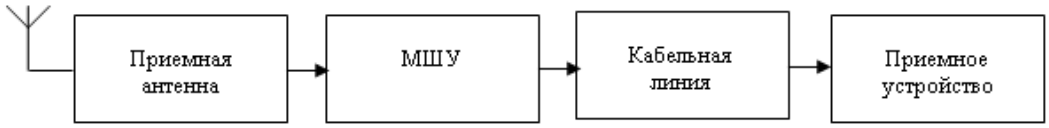
11.	Средняя мощность на входном полюсе волоконно-оптических линий передачи для одноканальных и многоканальных защищенных волоконно-оптических систем передачи (ВОСП) должна быть минимально возможной (чем меньше P, тем выше защищенность). Известны: коэффициент дополнительных потерь в волокне $k_д = 1$ (отн. ед.); коэффициент защищенности Z = 16,4 дБ; поправка для одноканальной защищенной волоконно-оптической системы передачи (ВОСП) U = 0 дБ. Определите, чему равно Ад [дБ], предельно допустимое изменение коэффициента передачи между оптическими полюсами. Запишите ответ по правилу округления до десятого знака после запятой.	5
Решение	Определим, чему равно предельно допустимое изменение коэффициента передачи между оптическими полюсами Ад: Ад = $(log_{10}\ (1 - \ 10^{\frac{Z}{-10}}))$ / (− 0,1) = 0,100648 ≈ 0,1дБ. **Проверка:** Ад = 0,1, тогда $Z = -10 * lg\ (1 - \ 10^{-0,1*А_д})$ = $-10 * lg\ (1 - \ 10^{-0,1*0,1}) = 16,427747 \approx$ 16,4дБ. **Ответ: Ад = 0,14дБ.**	
12.	Имеется приемная система, состоящая из антенны, (длинного и короткого) коаксиального кабеля, потерями в котором можно пренебречь, малошумящего усилителя и приемного устройства. Известны коэффициенты шума приемной системы: $К_{шКАБ} = 10$дБ, $К_{шМШУ} = 4$дБ, $К_{шПРМ} = 12$дБ; коэффициенты усиления приемной системы $К_{уКАБ} = -10$дБ, $К_{уМШУ} = 30$дБ. Рассматриваются 4 варианта построения системы: 1. антенна соединена с приемником коротким кабелем; 2. антенна соединена с приемником длинным кабелем; 3. антенна соединена с МШУ длинным кабелем, а МШУ с приемником коротким; 4. антенна соединена с МШУ коротким кабелем, а МШУ с приемником длинным. Определите, на сколько будет повышена чувствительность системы в 4-ом варианте относительно 3-го варианта (Кш3 - Кш4).	5

Решение

1) При первом варианте коэффициент шума системы равен коэффициенту шума приемника 12 дБ или 15,849



2) При четвертом варианте получаем

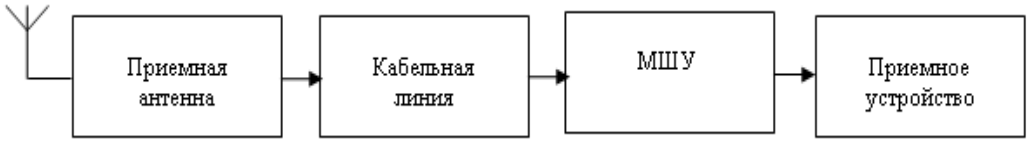


$$К_{ш} = К_{шМШУ} + \frac{К_{шКАБ}-1}{К_{уМШУ}} + \frac{К_{шПРМ}-1}{К_{уМШУ}\cdot К_{уКАБ}} = 2{,}512 + \frac{10-1}{1000} + \frac{15{,}849-1}{0{,}1\cdot 1000} = 2{,}67, \text{ раз}$$

или

$$К_{ш} = 10 \cdot log(2{,}67) = 4{,}264, \text{ дБ.}$$

3) При третьем варианте получаем



$$К_{ш} = К_{шКАБ} + \frac{К_{шМШУ}-1}{К_{уКАБ}} + \frac{К_{шПРМ}-1}{К_{уМШУ}\cdot К_{уКАБ}} = 10 + \frac{2{,}512-1}{0{,}1} + \frac{15{,}849-1}{0{,}1\cdot 1000} = 25{,}27, \text{ раз}$$

или

$$К_{ш} = 10 \cdot log(25{,}27) = 14{,}00 \text{ , дБ.}$$

Таким образом, чувствительность системы в четвертом варианте повышена относительно третьего варианта на

$$К_{ш}3 - К_{ш}4 = 14 - 4{,}264 = 9{,}736, \text{ дБ}$$

Ответ: 9,736 дБ