Критерии оценивания заданий заключительного этапа по направлению «Кибербезопасность»

Стеганография и цифровые водяные знаки

Критерии оценивания

За наличие идеи или направления, потенциально ведущего к решению, но без корректного решения и ответа — 1–2 балла;
За частично верный ответ — 5 баллов;
За описание корректного хода решения в неполном виде — 5 баллов (может суммироваться с прошлым пунктом при условии получения части ответа);
За полное, корректное и логически выстроенное решение с получением правильного ответа — 20 баллов.

Перед вами R-, G- и B-каналы цифрового изображения, подвергнутого атаке «Соль и Перец». Известно, что до атаки в одном из каналов содержалось сообщение, встроенное распространенным стеганографическим методом. Восстановите сообщение. Представьте ответ в виде осмысленного текста на русском языке, используя приведенную таблицу для декодирования

R-канал:

166	184	199	212	222	222	222	212	199	185	166	142	117	103	109	134
168	186	201	215	222	233	222	215	201	187	169	144	118	103	108	132
166	184	199	0	222	222	222	213	199	184	166	142	117	102	109	134
159	179	192	204	255	216	212	205	192	179	159	135	113	101	112	137
148	168	183	192	199	202	199	192	183	168	148	125	107	102	119	143
133	152	168	179	185	187	184	178	168	152	134	114	102	107	129	152
117	133	148	159	166	168	166	159	148	133	117	104	102	118	140	159
104	114	126	135	142	144	142	135	126	113	104	101	112	132	153	167
102	103	106	112	116	119	117	113	107	103	0	112	130	148	163	173

117	107	102	101	102	103	102	102	102	108	118	133	148	162	172	174
140	128	119	112	109	108	110	113	119	128	140	153	163	172	174	171
160	151	143	137	134	133	133	137	143	151	159	167	172	174	171	164
172	168	163	159	157	156	157	160	164	168	172	174	173	169	164	154
174	174	174	172	171	171	172	0	173	175	174	172	167	160	0	141
167	255	172	173	173	173	174	173	172	170	166	162	155	147	139	131
154	159	161	164	165	166	166	164	161	159	154	148	141	134	129	129

G-канал:

195	193	187	180	174	171	174	180	188	193	196	196	196	193	188	179
195	192	187	179	171	165	171	178	187	192	195	196	196	193	189	180
196	193	188	0	173	171	174	181	188	193	196	196	196	193	187	179
196	194	190	185	255	179	181	186	191	194	196	196	195	192	186	177
196	195	193	190	188	187	188	191	193	195	196	196	195	191	184	176
196	196	195	194	193	192	193	194	196	196	196	196	192	189	180	172
195	196	196	196	196	195	196	196	196	196	196	194	191	185	177	168
193	195	196	196	196	196	197	196	196	196	194	191	186	179	171	164
191	193	195	195	196	195	195	195	194	193	0	186	181	173	167	163
184	189	191	192	193	193	193	192	191	189	185	179	174	167	163	162
176	181	184	186	188	188	188	187	184	181	177	172	167	163	162	166
168	172	176	178	179	180	179	178	175	173	168	165	162	163	166	172
162	165	167	168	169	170	170	168	167	165	163	162	164	167	172	179

163	162	162	163	163	163	163	0	162	162	163	166	170	174	0	186
169	255	165	163	164	163	164	164	165	167	170	173	178	183	188	192
179	176	173	171	171	171	171	172	173	176	179	183	187	190	193	193

B-канал:

200	192	184	176	172	170	172	176	184	192	200	210	218	222	222	222
196	188	180	176	168	166	168	176	180	190	196	208	216	220	222	222
200	188	184	0	172	170	172	176	184	190	200	208	216	220	222	224
200	192	188	180	255	178	176	180	184	194	200	208	218	222	224	220
204	196	192	188	184	180	184	184	192	196	206	212	220	224	224	220
212	204	198	192	188	188	192	192	196	204	208	218	220	224	222	222
216	208	204	200	200	196	200	200	204	212	218	220	224	224	220	220
220	216	212	208	208	210	208	208	212	216	222	222	224	224	222	220
224	220	220	216	216	218	216	216	220	220	0	222	224	220	222	222
224	224	224	220	220	222	220	224	224	224	222	222	220	222	216	216
220	224	224	224	224	224	224	224	224	224	222	220	220	216	216	216
220	220	220	220	220	222	220	220	220	220	222	222	216	216	216	218
216	220	220	220	220	222	220	220	220	216	218	218	218	218	216	218
216	216	216	216	216	218	216	0	216	216	218	218	218	216	0	216
216	255	216	216	216	218	216	216	216	218	216	216	218	218	218	218
216	216	216	216	216	216	216	216	216	216	218	216	218	218	218	216

Кодировка:

А	00000100 00010000	Р	00000100 00100000	а	00000100 00101111	р	00000100 01000000	.	00000000 00101110
Б	00000100 00010001	С	00000100 00100001	б	00000100 00110000	с	00000100 01000001	,	00000000 00101100
В	00000100 00010010	Т	00000100 00100010	в	00000100 00110001	т	00000100 01000010	-	00000000 00101101
Г	00000100 00010011	У	00000100 00100011	г	00000100 00110010	у	00000100 01000011	–	00100000 00010011
Д	00000100 00010100	Ф	00000100 00100100	д	00000100 00110011	ф	00000100 01000100	$	00100000 00010100
Е	00000100 00010101	Х	00000100 00100101	е	00000100 00110100	х	00000100 01000101	?	00000000 00100100
Ж	00000100 00010110	Ц	00000100 00100110	ж	00000100 00110101	ц	00000100 01000110	!	00000000 00111111
З	00000100 00010111	Ч	00000100 00100111	з	00000100 00110110	ч	00000100 01000111	+	00000000 00100001
И	00000100 00011000	Ш	00000100 00101000	и	00000100 00110111	ш	00000100 01001000	×	00000000 00101011
Й	00000100 00011001	Щ	00000100 00101001	й	00000100 00111000	щ	00000100 01001001	÷	00000000 11010111
К	00000100 00011010	Ъ	00000100 00101010	к	00000100 00111001	ъ	00000100 01001010	=	00000000 11110111
Л	00000100 00011011	Ы	00000100 00101011	л	00000100 00111010	ы	00000100 01001011		00000000 00100000

М	00000100 00011100	Ь	00000100 00101100	м	00000100 00111011	ь	00000100 01001100		
Н	00000100 00011101	Э	00000100 00101101	н	00000100 00111100	э	00000100 01001101		
О	00000100 00011110	Ю	00000100 00101110	о	00000100 00111101	ю	00000100 01001110		
П	00000100 00011111	Я	00000100 00101111	п	00000100 00111110	я	00000100 01001111		

Решение:
Должно получится сообщение “Путь – где воля.”
В B-канале все пиксели кратны двум, так как в него встроено сообщение с использованием метода QIM и шага квантования 4. При построении гистограммы B-канала получились бы пробелы, свойственные методу QIM с этим шагом квантования и не встречающиеся в других базовых методах (LSB, HS, DE и т. д.). Исключение представляют пиксели, зашумленные атакой “Соль и Перец” и имеющие из-за этого значения 255, которые надо проигнорировать при определении способа встраивания. Для извлечения двоичного сообщения необходимо взять остаток от деления на 4 для каждого пикселя и, затем, используя таблицу с кодировкой и учитывая возможную инверсию значений из-за шума “Соль и Перед”, восстановить исходное сообщение.

Криптография

<u>Критерии оценивания</u>

Правильно определён шифр: 4 балла (2 – если без аргументов, 4 – если с аргументами; неправильно определён шифр, но есть логика в идее – 2 балла);

Правильно составлена ключевая матрица: 4 балла;

Правильно найдена обратная матрица: 6 баллов;

Правильно расшифровано сообщение на найденном ключе: 4 балла;

Правильный ответ на расшифрованный вопрос: 2 балла.

Итого максимально за задачу можно получить 20 баллов.

Задача:

Ответьте на вопрос «щыбйёттёинущьла», используя следующую подсказку:

В старом доме на холме

Кто-то помнит обо мне,

Разливает чай.

Поднимусь скорей на холм,

Отопру дверь в старый дом

С помощью ключа.

Расцветёт от счастья холм,

И покинет старый дом

Наконец печаль.

Решение:

Сообщение и подсказка представляют собой тексты на естественном языке, следовательно, речь идёт об исторических шифрах. Стихотворение представляет собой три трёхстишия (3×3). В каждом трёхстишии используется слово холм, холм по-английски hill, как и название одного из исторических шифров – Hill cipher. Отсюда следует, что для зашифрования использовался шифр Хилла с ключевой матрицей 3×3.

Чтобы расшифровать текст необходимо получить ключевую матрицу. Можно догадаться, что первая буква каждой строки представляет собой элемент ключевой матрицы, поскольку эти буквы заглавные, хотя в шифртексте нет заглавных букв. Если поставить каждой букве алфавита число от 0 до 32 (шифртекст содержит «ё», поэтому алфавит – все 33 русские буквы) в соответствие, то получится ключевая матрица . Поскольку , найденная матрица подходит в качестве ключевой матрицы шифра Хилла. Для расшифрования необходимо найти обратную к ней. Таковой матрицей будет . Используя данную матрицу, получим расшифрованный текст «дешифрованиеэто».

В конце решения необходимо дать ответ на расшифрованный вопрос. Формулировка ответа может быть произвольной, но из неё должно следовать, что дешифрование – это раскрытие зашифрованной информации методами криптоанализа (без знания ключа). Если из ответа следует, что дешифрование и расшифрование – это синонимы, то ответ неправильный.

Критерии оценивания

Правильно определён шифр: 4 балла (2 – если без аргументов, 4 – если с аргументами; неправильно определён шифр, но есть логика в идее – 2 балла);

Правильно составлена ключевая матрица: 4 балла;

Правильно найдена обратная матрица: 6 баллов;

Правильно расшифровано сообщение на найденном ключе: 4 балла;

Правильный ответ на расшифрованный вопрос: 2 балла.

Итого максимально за задачу можно получить 20 баллов.

Безопасная разработка

Критерии оценивания:
- 0 баллов — ответ не представлен или не имеет отношения к заданию

- 1-5 баллов — ответ представлен, но не решает поставленную задачу, ход решения неверный
- 6-12 баллов — ответ представлен, ход решения верный, однако не учтены фундаментальные ограничения задачи или языка программирования
- 13-16 баллов — ответ представлен, ход решения верный, не учтены только требования, касающиеся кроссплатформенности представленного решения
- 16-18 баллов — ответ представлен, ход решения верный, все требования учтены

+ 2 балла — представлены рабочие unit-тесты для решения

Существует программное обеспечение для работы с пользователями некоторой информационной системы.

Класс пользователя объявлен следующим образом:

```
class User
{
    unsigned char m_age;
    char * m_login;
    char * m_password;

public:
    User(unsigned char age, char * login, char * password);
    virtual ~User() = default;

    /**
     * @brief Возвращает возраст пользователя
     */
    unsigned char get_age() const;

    /**
     * @brief Проверяет, равен ли логин пользователя переданному
     *
     * @param login Логин для проверки
     * @returns `true` если логин совпадает, `false` иначе
     */
    bool check_login(char const * login) const;

    /**
     * @brief Проверяет, равен ли пароль пользователя переданному
```

```
     *
     * @param password Пароль для проверки
     * @returns `true` если пароль совпадает, `false` иначе
     */
    bool check_password(char const * password) const;
};
```

Официально вашей задачей в рамках работы с данным ПО является разработка простой функции, проверяющей, достиг ли переданный пользователь возраста 18 лет. Если возраст пользователя 18 лет или старше, функция должна вернуть `true`, в противном случае функция должна вернуть `false`. Функция должна быть объявлена следующим образом:

```
bool check_age(User const& user)
```

Однако в рамках данного задания вам необходимо, исполняя роль злоумышленника, сделать так, чтобы функция имела следующий побочный эффект. Всем объектам пользователей, которые будут переданы на проверку данной функции, должен безусловно задаваться пароль “`qwerty1234567890`”. Гарантируется, что в функцию передаются неконстантные объекты.

Во время работы над ПО у вас нет прав на изменение программного кода класса `User`, а также на любые другие манипуляции, помимо разработки функции `check_age`. Разрешается только создавать вспомогательные функции при необходимости. При этом гарантируется, что конструктор класса корректно инициализирует поля в соответствии с их названиями. Логин и пароль не шифруются.

Необходимо учитывать, что у вас нет каких-либо достоверных предположений о том, на какой платформе будет компилироваться и запускаться данное ПО, равно как и о каких-либо параметрах обрабатываемых в нём данных.

В качестве ответа следует предоставить программный код решения, а также unit-тесты, проверяющие его работоспособность для различных условий. В целях тестирования разрешается представить собственную реализацию методов класса `User` (только тех, которые уже объявлены в представленном фрагменте кода).

Безопасность компьютерных сетей и систем

Ключи

А) 5: 78.12.34.56, 91.234.56.78, 154.32.45.67, 198.51.100.200, 198.162.3.4 Б) 5 IP: 45.67.89.123, 123.45.67.89, 203.0.113.1, 192.0.2.1, 112.218.136.6

В) Пример блокирующих правил: /ip firewall address-list add list=TARPIT_BAN address=78.12.34.56 timeout=1d comment="Сканер tarpit" add list=TARPIT_BAN address=91.234.56.78 timeout=1d comment="Сканер tarpit" #add list=TARPIT_BAN address=99.88.77.66 timeout=1d comment="Сканер tarpit" add list=TARPIT_BAN address=154.32.45.67 timeout=1d comment="Сканер tarpit" add list=TARPIT_BAN address=198.51.100.200 timeout=1d comment="Сканер tarpit" add list=TARPIT_BAN address=198.162.3.4 timeout=1d comment="Сканер tarpit" #Блокирующее правило для DROP. /ip firewall filter add chain=input action=drop src-address-list=TARPIT_BAN \ log=yes log-prefix="BLOCKED_SCANNER" place-before=0 comment="Блок tarpitсканеров"

Вместо примера выше, могут быть семантически корректные описанные правила.

Детализация критериев:

А) Количество IP адресов: 5 (78.12.34.56, 91.234.56.78, 154.32.45.67, 198.51.100.200, 198.162.3.4), баллы: 7.

А.1) С текстовым объяснением допускается и наличие 6 IP-адреса: 99.88.77.66. При наличии IP-адреса 99.88.77.66 опционально, но с подозрением на ИИ), баллы: 7.

А.2) При наличии 4 верных IP-адресов из списка, баллы 4.

А.3) Количество IP адресов <4 или адреса указаны неверно, последовательность значения не имеет, баллы: 0-2.

Б) Количество IP адресов: 5 (45.67.89.123, 123.45.67.89, 203.0.113.1, 192.0.2.1, 112.218.136.6), баллы 7.

Б.1) При наличии 5 IP-адресов или наличие логики с объяснением, баллы 7.

Б.2) При наличии 4 IP-адресов, баллы 5.

Б.3) Количество IP адресов <4 или адреса указаны частично неверно, баллы: 0-2.
В) Полный список адресов из пункта А и корректная логика, либо синтаксис, либо семантика, баллы 7.

B.1) Частичная реализация или неверный список IP-адресов из пункта А, баллы 3-4.

B.2) Нет верных IP-адресов или неверная логика, баллы 0-2.

B.3) Наличие блокирующего правила, 1 балл (входит в оценку 7), но наличие блокирующего правила может оцениваться отдельно.

Текстовый вариант:

Проанализируйте лог файл

** АНАЛИЗ ЛОГОВ MIKROTIK FIREWALL**

Сессия #47 | 11.02.2026 03:15:42-14:56:00

03:15:42 firewall,info prerouting: ... 99.88.77.66:11111->100.299.203.100:8080

03:15:42 firewall,info EMPIRE_TARPIT input: ... 99.88.77.66:11111->100.299.203.100:8080

04:22:15 firewall,info KNOCK1 input: ... 45.67.89.123:54321->100.299.203.100:7000

04:22:15 firewall,info prerouting: ... 45.67.89.123:54321->100.299.203.100:8080

04:22:15 firewall,info prerouting: ... 100.299.203.100:8080->45.67.89.123:54321 [RESPONSE]

04:22:15 firewall,info KNOCK1 input: ... 45.67.89.123:54321->100.299.203.100:7000

04:22:15 firewall,info EMPIRE_TARPIT input: ... 45.67.89.123:54321->100.299.203.100:8080

04:13:39 firewall,info prerouting: in:ether1 ... 112.218.136.6:45958->100.299.203.100:8080

05:33:47 firewall,info prerouting: ... 78.12.34.56:67890->100.299.203.100:8080

05:33:47 firewall,info EMPIRE_TARPIT input: ... 78.12.34.56:67890->100.299.203.100:8080

05:33:47 firewall,info prerouting: ... 100.299.203.100:8080->78.12.34.56:67890 [RESPONSE]

07:19:22 firewall,info prerouting: ... 91.234.56.78:11223->100.299.203.100:8080

07:19:22 firewall,info EMPIRE_TARPIT input: ... 91.234.56.78:11223->100.299.203.100:8080

07:19:22 firewall,info prerouting: ... 100.299.203.100:8080->91.234.56.78:11223 [RESPONSE]

08:44:19 firewall,info prerouting: ... 123.45.67.89:33445->100.299.203.100:8080

08:44:19 firewall,info nat: SNAT ... 123.45.67.89:33445->100.299.203.100:8080

08:44:19 firewall,info mangle: mark-packet ... 123.45.67.89:33445->100.299.203.100:8080

08:44:19 firewall,info input: ... 123.45.67.89:33445->100.299.203.100:8080

08:44:19 firewall,info KNOCK1 input: ... 123.45.67.89:33445->100.299.203.100:7000

09:15:32 firewall,info prerouting: ... 45.67.89.123:44567->100.299.203.100:8080 # тот же IP!

09:15:32 firewall,info prerouting: ... 45.67.89.123:44568->100.299.203.100:7000 # другой порт

09:15:32 firewall,info EMPIRE_TARPIT input: ... 45.67.89.123:44567->100.299.203.100:8080

10:01:23 firewall,info prerouting: ... 154.32.45.67:12345->100.299.203.100:8081

10:01:23 firewall,info EMPIRE_TARPIT input: ... 154.32.45.67:12345->100.299.203.100:8081

11:22:44 firewall,info prerouting: ... 203.0.113.1:98765->100.299.203.100:8080

11:22:44 firewall,info KNOCK1 input: ... 203.0.113.1:98765->100.299.203.100:7000

12:33:51 firewall,info prerouting: ... 198.51.100.200:54321->100.299.203.100:8080

12:33:51 firewall,info EMPIRE_TARPIT input: ... 198.51.100.200:54321->100.299.203.100:8080

13:44:66 firewall,info prerouting: ... 192.0.2.1:11111->100.299.203.100:8082

13:44:66 firewall,info KNOCK1 input: ... 192.0.2.1:11111->100.299.203.100:7000

14:55:77 firewall,info prerouting: ... 198.162.3.4:22222->100.299.203.100:8080

14:55:77 firewall,info EMPIRE_TARPIT input: ... 198.162.3.4:22222->100.299.203.100:8080

А) Сколько уникальных IP-адресов попало в ‘tarpit’ до KNOCK1? Перечислите IP-адреса.

Б) Сколько IP адресов не попало в ‘tarpit’ до KNOCK1? Перечислите IP-адреса.

В) "Интегрировать IP-адреса из п. А) в систему защиты в виде нового блокирующего правила на примере TARPIT и Port Knocking.

Представьте развернутый ответ с семантикой (указанием списка адресов и реализацией).

В виде рисунка

А) Сколько уникальных IP-адресов попало в ‘tarpit’ до KNOCK1? Перечислите IP-адреса.

Б) Сколько IP адресов не попало в ‘tarpit’ до KNOCK1? Перечислите IP-адреса.

В) "Интегрировать IP-адреса из п. А) в систему защиты в виде нового блокирующего правила на примере TARPIT и Port Knocking.

Представьте развернутый ответ с семантикой (указанием списка адресов и реализацией).

Безопасность систем искусственного интеллекта

КРИТЕРИИ

Часть 1. Атакующий (макс. 10 баллов)

За наличие идеи поиска аномалий без корректного метода: 1–2 балла; за частично верное решение (найдена часть канареек с обоснованием): 3-8 баллов; за полное решение с нахождением всех 5 канареек, обоснованием и верным очищенным датасетом: 10 баллов. Штраф за валидные записи ошибочно помеченные как канарейки.

Часть 2. Защитник (макс. 10 баллов)

За наличие идеи или поверхностный анализ кода без нахождения конкретных ошибок: 1–2 балла; за частично верный анализ (найдены некоторые ошибки с обоснованием): 3-8 баллов; за полный анализ с нахождением основных ошибок, объяснением их последствий и конкретными исправлениями: 10 баллов.

Международный радиоастрономический консорциум ведёт проект по поиску внеземного разума – программу «Орион». Ядро проекта – ИИ-платформа, которая автоматически классифицирует радиосигналы, принятые глобальной сетью телескопов: отделяет пульсарный шум, техногенные помехи и атмосферные аномалии от потенциальных «кандидат-сигналов» внеземного происхождения. Платформа ведёт реестр размеченных обучающих примеров, на которых регулярно дообучается модель.

Датасет «Ориона» – результат нескольких лет наблюдений и ручной разметки десятками операторов; это уникальная интеллектуальная собственность консорциума. После подозрения на несанкционированный доступ к обучающему датасету (возможная кража для обучения модели конкурирующим проектом), младший инженер по безопасности ИИ-систем (позывной «Сокол») внедрил в реестр фиктивные записи – «сэмплы-канарейки» (canary samples). Идея в том, что если злоумышленник украдёт датасет и обучит на нём собственную модель, наличие канареек можно будет подтвердить с помощью membership inference attack – это станет доказательством кражи данных.

«Сокол» покинул проект, не оставив документации. Сколько канареек он внедрил и как их отличить – неизвестно.

Вы – аналитик по безопасности ИИ-систем. Вам передали фрагмент реестра и исходный код «Сокола». Задача: очистить датасет от канареек перед следующим циклом дообучения, чтобы не допустить отравления модели (data poisoning) некорректными данными.

Из технической документации платформы известно:

1. Идентификаторы обучающих примеров имеют формат , где – цифры 0–9, а C – контрольный символ [0-9X].

2. Контрольный символ вычисляется так:

Весовые коэффициенты — первые четыре простых числа в порядке возрастания.

3. Все обучающие примеры загружаются в систему строго по расписанию обсерваторского конвейера данных (data pipeline): по вторникам, четвергам и субботам. За время проекта ни одной внеплановой загрузки не было.

4. Поле «Приоритет» выставляется автоматически по «классификационному протоколу»: каждому типу сигнала соответствует ровно один допустимый уровень приоритета (соответствие не раскрывается во внешней документации).

Календарь на март 2025

Пн	Вт	Ср	Чт	Пт	Сб	Вс
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31						

Часть 1. Атакующий

Реестр обучающих примеров (фрагмент, март 2025)

№	ID сэмпла	Оператор	Тип сигнала	Дата загрузки	Объём (фреймов)	Приоритет
1	OR-7412-1	Волков Д.А.	Космический фон	25.03.2025	45	Рутинный

2	OR-4738-2	Савицкая Е.О.	Пульсарный шум	08.03.2025	30	Рутинный
3	OR-1523-4	Волков Д.А.	Космический фон	04.03.2025	45	Рутинный
4	OR-5300-8	Чен Ли	Техногенная помеха	19.03.2025	20	Интересный
5	OR-9351-4	Рамирес М.К.	Кандидат-сигнал	20.03.2025	50	Аномальный
6	OR-3871-6	Чен Ли	Атмосферная аномалия	06.03.2025	60	Интересный
7	OR-6245-7	Рамирес М.К.	Кандидат-сигнал	18.03.2025	55	Аномальный
8	OR-9162-8	Чен Ли	Атмосферная аномалия	13.03.2025	25	Интересный
9	OR-2956-X	Савицкая Е.О.	Пульсарный шум	15.03.2025	40	Рутинный
10	OR-5849-6	Рамирес М.К.	Кандидат-сигнал	22.03.2025	55	Аномальный
11	OR-2173-8	Волков Д.А.	Техногенная помеха	11.03.2025	35	Интересный

12	OR-7200-9	Савицкая Е.О.	Пульсарный шум	27.03.2025	35	Аномальный
13	OR-8360-0	Савицкая Е.О.	Пульсарный шум	29.03.2025	30	Рутинный

Вопросы

1.1 Определите все канареечные записи в реестре. Для каждой обоснуйте, почему она является канарейкой, и покажите ход рассуждений.

1.2 Выпишите ID сэмплов восстановленного (очищенного) датасета.

Часть 2. Защитник

Вам удалось получить код, который «Сокол» использовал для генерации канареек:

```
import random

OPERATORS = ["Волков Д.А.", "Савицкая Е.О.", "Чен Ли", "Рамирес М.К."]

SIGNAL_TYPES = [
"Пульсарный шум",
"Техногенная помеха",
"Кандидат-сигнал",
"Атмосферная аномалия",
    "Космический фон"
]

PRIORITIES = ["Рутинный", "Интересный", "Аномальный"]

```

```
def compute_check_char(d1, d2, d3, d4):
    return (d1 + d2 + d3 + d4) % 10

def generate_sample_id():
    digits = [random.randint(0, 9) for _ in range(4)]
    check = compute_check_char(*digits)
    return f"OR-{'.'.join(map(str, digits))}-{check}"

def generate_canary():
day = random.randint(1, 31)
    record = {
        "id":       generate_sample_id(),
        "operator": random.choice(OPERATORS),
        "signal":   random.choice(SIGNAL_TYPES),
        "date":     f"{day:02d}.03.2025",
        "frames":   random.randint(10, 60),
        "priority": random.choice(PRIORITIES)
}
    return record

def inject_canaries(real_data, n_canaries):
    random.seed(1337)
    result = list(real_data)
for _ in range(n_canaries):
        canary = generate_canary()
        result.append(canary)
        print(f"Injected canary: {canary['id']}")
    random.shuffle(result)
    return result
```

Вопросы

2.1 Перечислите ошибки в коде «Сокола», которые позволяют обнаружить канарейки или снижают качество их маскировки. Для каждой ошибки укажите номер строки, суть проблемы и объясните, к какому методу обнаружения она приводит.

2.2 Предложите исправления для каждой найденной ошибки. Можно использовать псевдокод, фрагменты кода или подробное описание.

РЕШЕНИЕ

Часть 1. Атакующий

1.1 Выявление канареечных записей

Проверка контрольного символа

Формат идентификатора: OR-$d_1d_2d_3d_4$-C. Весовые коэффициенты – первые четыре простых числа: 2, 3, 5, 7. Контрольный символ вычисляется как $S = (2 \cdot d_1 + 3 \cdot d_2 + 5 \cdot d_3 + 7 \cdot d_4) \bmod 11$. При S = 10 записывается символ X, в остальных случаях – цифра S.

Результаты вычислений для всех записей реестра:

№1: OR-7412-1. 14+12+5+14 = 45. 45 mod 11 = 1. Указан 1 – совпадает.

№2: OR-4738-2. 8+21+15+56 = 100. 100 mod 11 = 1. Указан 2 – не совпадает. Канарейка.

№3: OR-1523-4. 2+15+10+21 = 48. 48 mod 11 = 4. Указан 4 – совпадает.

№4: OR-5300-8. 10+9+0+0 = 19. 19 mod 11 = 8. Указан 8 – совпадает.

№5: OR-9351-4. 18+9+25+7 = 59. 59 mod 11 = 4. Указан 4 – совпадает.

№6: OR-3871-6. 6+24+35+7 = 72. 72 mod 11 = 6. Указан 6 – совпадает.

№7: OR-6245-7. 12+6+20+35 = 73. 73 mod 11 = 7. Указан 7 – совпадает.

№8: OR-9162-8. 18+3+30+14 = 65. 65 mod 11 = 10. Ожидается X, указан 8 – не совпадает. Канарейка.

№9: OR-2956-X. 4+27+25+42 = 98. 98 mod 11 = 10. Ожидается X, указан X – совпадает. Запись валидна; значение X здесь закономерно и подтверждает корректность формулы.

№10: OR-5849-6. 10+24+20+63 = 117. 117 mod 11 = 7. Указан 6 – не совпадает. Канарейка.

№11: OR-2173-8. 4+3+35+21 = 63. 63 mod 11 = 8. Указан 8 – совпадает.

№12: OR-7200-9. 14+6+0+0 = 20. 20 mod 11 = 9. Указан 9 – совпадает.

№13: OR-8360-0. 16+9+30+0 = 55. 55 mod 11 = 0. Указан 0 – совпадает.

По данному критерию выявлены канарейки: №2, №8, №10.

Проверка дня недели

Согласно документации, загрузка данных производится строго по вторникам, четвергам и субботам. Внеплановых загрузок не зафиксировано.

Допустимые даты загрузки в марте 2025 года: 1 (сб), 4 (вт), 6 (чт), 8 (сб), 11 (вт), 13 (чт), 15 (сб), 18 (вт), 20 (чт), 22 (сб), 25 (вт), 27 (чт), 29 (сб).

При проверке всех записей обнаруживается одно нарушение: запись №4 имеет дату загрузки 19.03.2025, что соответствует среде. Среда не входит в число допустимых дней. Остальные записи загружены в допустимые дни.

По данному критерию выявлена канарейка: №4.

Проверка соответствия «тип сигнала – приоритет»

По документации, каждому типу сигнала соответствует ровно один уровень приоритета. Таблицу соответствий можно восстановить по записям, не отмеченным как канарейки на предыдущих этапах:

Космический фон – Рутинный (подтверждается записями №1, №3).

Пульсарный шум – Рутинный (подтверждается записями №9, №13).

Техногенная помеха – Интересный (подтверждается записью №11).

Атмосферная аномалия – Интересный (подтверждается записью №6).

Кандидат-сигнал – Аномальный (подтверждается записями №5, №7).

При проверке всех записей по восстановленной таблице обнаруживается одно нарушение: запись №12 имеет тип «Пульсарный шум» и приоритет «Аномальный», тогда как для данного типа сигнала установлен приоритет «Рутинный».

По данному критерию выявлена канарейка: №12.

Обнаружено пять канареечных записей:

№2 (OR-4738-2) – несовпадение контрольного символа (ожидается 1, указан 2).

№4 (OR-5300-8) – загрузка в среду, что противоречит расписанию (вт/чт/сб).

№8 (OR-9162-8) – несовпадение контрольного символа (ожидается X, указан 8).

№10 (OR-5849-6) – несовпадение контрольного символа (ожидается 7, указан 6).

№12 (OR-7200-9) – нарушение соответствия типа сигнала и приоритета (пульсарный шум – аномальный вместо рутинного).

1.2 Очищенный датасет

После удаления канареек в реестре остаётся восемь валидных записей:

OR-7412-1, OR-1523-4, OR-9351-4, OR-3871-6, OR-6245-7, OR-2956-X, OR-2173-8, OR-8360-0.

Часть 2. Защитник

2.1 Ошибки в коде

Ошибка 1. Неверная формула контрольного символа (строка 16).

В коде используется выражение (d1 + d2 + d3 + d4) % 10 – простая сумма цифр по модулю 10. Согласно документации платформы, корректная формула предполагает взвешенную сумму с коэффициентами 2, 3, 5, 7 по модулю 11. Вследствие этого идентификаторы канареек не проходят валидацию по настоящей формуле, что и позволило обнаружить записи №2, №8 и №10.

Ошибка 2. Генерация произвольного дня месяца без учёта расписания (строка 24).

Выражение random.randint(1, 31) выбирает день равномерно из всего диапазона, не ограничиваясь допустимыми днями загрузки (вторник, четверг, суббота). В результате канарейка может получить дату, приходящуюся на недопустимый день недели. Обнаруживается проверкой дат по календарю – именно так была выявлена запись №4.

Ошибка 3. Случайный выбор приоритета (строка 31).

Выражение random.choice(PRIORITIES) назначает приоритет случайно из трёх вариантов, тогда как в реальной системе между типом сигнала и приоритетом существует однозначное соответствие. Это порождает невозможные сочетания. Так была выявлена запись №12.

Ошибка 4. Фиксированный seed генератора случайных чисел (строка 36).

Вызов random.seed(1337) делает генерацию полностью детерминированной. При наличии исходного кода (или при подборе seed перебором типичных значений) злоумышленник может воспроизвести все канарейки и удалить их из украденного датасета, что полностью обесценивает защитный механизм.

Ошибка 5. Невозможность генерации символа X (строка 16).

Оператор mod 10 возвращает значения от 0 до 9; значение 10 недостижимо. В настоящей формуле (mod 11) значение S = 10 преобразуется в символ X. Следовательно, ни одна канарейка не может иметь контрольный символ X, что создаёт статистически обнаруживаемую аномалию в распределении контрольных символов.

Ошибка 6. Вывод идентификаторов канареек в стандартный поток (строка 41).

Выражение print(f"Injected canary: {canary['id']}") записывает идентификатор каждой внедрённой канарейки в stdout. При сохранении логов (что является стандартной практикой) любое лицо, имеющее к ним доступ, получает полный список канареек без необходимости анализа данных.

Ошибка 7. Потенциальная генерация несуществующих дат (строка 24).

Выражение random.randint(1, 31) может породить дату, не существующую в некоторых месяцах (например, 30 или 31 февраля). В случае марта проблема не проявляется (в марте 31 день), однако код не является универсальным и при применении к другим месяцам может привести к ошибке или очевидно фиктивной записи.

2.2 Исправления

К ошибкам 1 и 5 (формула и символ X). Следует заменить тело функции compute_check_char на корректную формулу:

```
def compute_check_char(d1, d2, d3, d4):
    s = (2*d1 + 3*d2 + 5*d3 + 7*d4) % 11
    return 'X' if s == 10 else str(s)
```

Это одновременно устраняет неверные веса и модуль, а также обеспечивает возможность генерации символа X.

К ошибкам 2 и 7 (даты). Вместо случайного дня из фиксированного диапазона необходимо формировать список допустимых дат (вторник, четверг, суббота) для заданного месяца и выбирать из него:

```
from datetime import date
import calendar
def valid_upload_days(year, month):
    return [d for d in range(1, calendar.monthrange(year, month)[1] + 1)
    if date(year, month, d).weekday() in (1, 3, 5)]
day = random.choice(valid_upload_days(2025, 3))
```

Это также исключает генерацию несуществующих дат, поскольку monthrange возвращает фактическое количество дней в месяце.

К ошибке 3 (приоритет). Необходимо заменить случайный выбор приоритета на детерминированное соответствие:

```
PRIORITY_MAP = {
        "Космический фон":          "Рутинный",
        "Пульсарный шум":           "Рутинный",
        "Техногенная помеха":   "Интересный",
        "Атмосферная аномалия": "Интересный",
   "Кандидат-сигнал":      "Аномальный",
}
priority = PRIORITY_MAP[signal]
```

К ошибке 4 (seed). Следует отказаться от фиксированного seed. Можно либо не инициализировать генератор явно (в этом случае используется системная энтропия), либо применить криптографически стойкий генератор secrets.SystemRandom(). Если воспроизводимость необходима для целей тестирования, seed должен храниться в защищённом хранилище с ограниченным доступом, а не быть захардкожен в исходном коде.

К ошибке 6 (print). Строку с print следует удалить. При необходимости аудита информацию о внедрённых канарейках следует записывать в зашифрованный журнал с ограниченным доступом.