

# Содержание

<b>Введение</b>	<b>17</b>
<b>Предисловие</b>	<b>19</b>
<b>Об авторе</b>	<b>25</b>
<b>Глава 1. Бег с ножницами</b>	<b>27</b>
1.1. Измерение угрозы	31
Как подсчитать стоимость	31
Кто несет угрозу	33
Безопасность программного обеспечения	34
1.2. Концепции безопасности	36
Стратегия безопасности	37
Недостатки безопасности	38
Уязвимости	38
Использование уязвимостей	39
Контрмеры	40
1.3. С и С++	41
Краткая история	42
Проблемы языка программирования С	43
Старый код	46
Другие языки	46
1.4. Платформы разработки	47
Операционные системы	48
Компиляторы	48
1.5. Резюме	48
1.6. Дополнительная литература	49
<b>Глава 2. Строки</b>	<b>51</b>
2.1. Символьные строки	51
Строковый тип данных	52
UTF-8	54
Широкие строки	55
Строковые литералы	55
Строки в С++	57
Символьные типы	58
Размеры строк	60
2.2. Распространенные ошибки при работе со строками	62
Некорректно ограниченные строки	62
Ошибки сдвига на единицу	66
Ошибки, связанные с нулевым завершающим символом	67
Усечение строк	68
Строковые ошибки без функций	69

2.3. Уязвимости, связанные со строками, и их использование	69
Ненадежные данные	70
Недостаток безопасности: IsPasswordOK	71
Переполнение буфера	72
Организация памяти процесса	73
Управление стеком	74
Разрушение стека	77
Внедрение кода	81
Внедрение дуги	86
Возврат-ориентированное программирование	87
2.4. Стратегии контрмер при работе со строками	88
Обработка строк	89
Интерфейсы C11 проверки выхода за границы	89
Функции динамического выделения памяти	92
C++ <code>std::basic_string</code>	95
Недействительность ссылок на строковые объекты	96
Прочие распространенные ошибки при использовании <code>basic_string</code>	98
2.5. Функции для работы со строками	98
<code>gets()</code>	98
C99	99
Интерфейсы из приложения К стандарта C11: <code>gets_s()</code>	101
Функции динамического распределения памяти	101
<code>strcpy()</code> и <code>strcat()</code>	103
C99	103
<code>strncpy()</code> и <code>strncat()</code>	106
<code>memcpy()</code> и <code>memmove()</code>	111
<code>strlen()</code>	112
2.6. Стратегии защиты времени выполнения	113
Обнаружение и восстановление	113
Проверка входных данных	113
Проверка размера объектов	114
Проверки времени выполнения, генерируемые компилятором Visual Studio	117
Стековые “канарейки”	119
Защита от разрушения стека (ProPolice)	120
Стратегии операционных систем	122
Обнаружение и восстановление	122
Неисполнимые стеки	123
W^X	124
PaX	125
Будущие направления	126
2.7. Широко известные уязвимости	126
Удаленный вход	127
Kerberos	127
2.8. Резюме	128
2.9. Дополнительная литература	129

<b>Глава 3. Уловки с указателями</b>	<b>131</b>
3.1. Местоположение данных	132
3.2. Указатели на функции	133
3.3. Указатели на объекты	134
3.4. Модификация указателя инструкции	135
3.5. Глобальная таблица смещений	137
3.6. Раздел <code>.ctors</code>	138
3.7. Виртуальные указатели	140
3.8. Функции <code>atexit()</code> и <code>on_exit()</code>	141
3.9. Функция <code>longjmp()</code>	142
3.10. Обработка исключений	144
Структурная обработка исключений	144
Системная обработка исключений по умолчанию	146
3.11. Стратегии противодействия	147
Стековые канарейки	147
<code>W^X</code>	147
Кодирование и декодирование указателей на функции	147
3.12. Резюме	149
3.13. Дополнительная литература	149
<b>Глава 4. Управление динамической памятью</b>	<b>151</b>
4.1. Управление памятью в языке программирования C	152
Стандартные функции управления памятью в C	152
Выравнивание	153
<code>alloca()</code> и массивы переменной длины	155
4.2. Распространенные ошибки управления памятью в C	156
Ошибки инициализации	156
Отсутствие проверки возвращаемых значений	158
Разыменованые нулевых или неверных указателей	160
Обращение к освобожденной памяти	161
Многократное освобождение памяти	162
Утечки памяти	162
Выделение памяти нулевого размера	163
DR #400	165
4.3. Управление динамической памятью в C++	166
Функции распределения памяти	167
Функции освобождения памяти	171
Сборка мусора	172
4.4. Распространенные ошибки управления памятью в C++	174
Некорректная обработка сбоев выделения памяти	174
Использование не соответствующих функций выделения и освобождения памяти	175
Многократное освобождение памяти	178
Функция освобождения памяти, генерирующая исключение	181
4.5. Диспетчеры памяти	181

4.6. Распределитель памяти Дуга Ли	183
Переполнения буфера в куче	185
4.7. Уязвимость, связанная с двойным освобождением памяти	191
Запись в освобожденную память	195
RtlHeap	195
Возврат к переполнению буфера	201
4.8. Стратегии противодействия	208
Нулевые указатели	208
Последовательные соглашения по управлению памятью	209
phkmalloс	209
Рандомизация	211
OpenBSD	211
Диспетчер памяти jemalloс	212
Статический анализ	212
Инструменты анализа времени выполнения	214
4.9. Знаменитые уязвимости	217
Уязвимость переполнения буфера в CVS	217
Microsoft Data Access Components (MDAC)	217
Повторное освобождение памяти в сервере CVS	218
Уязвимости в MIT Kerberos 5	218
4.10. Резюме	219

## **Глава 5. Целочисленная безопасность** **221**

5.1. Введение в целочисленную безопасность	221
5.2. Целочисленные типы данных	222
Беззнаковые целочисленные типы	223
Циклический возврат	224
Знаковые целочисленные типы	227
Диапазоны знаковых целых чисел	230
Целочисленное переполнение	232
Символьные типы	234
Модели данных	234
Другие целочисленные типы	235
5.3. Целочисленные преобразования	239
Преобразование целых чисел	239
Ранг целочисленного преобразования	239
Целочисленное повышение	240
Обычные арифметические преобразования	241
Преобразования из беззнаковых целочисленных типов	242
Преобразования из знаковых целочисленных типов	245
Следствия преобразований	248
5.4. Целочисленные операции	248
Присваивание	249
Сложение	251
Вычитание	257
Умножение	259

Деление и получение остатка	263
Унарный минус	267
Сдвиги	267
5.5. Уязвимости, связанные с целыми числами	270
Уязвимости	270
Циклический возврат	270
Ошибки преобразования и усечения	272
Целочисленные логические ошибки	274
5.6. Стратегии контрмер	274
Выбор целочисленного типа	275
Абстрактные типы данных	277
Арифметика произвольной точности	278
Проверка диапазона	279
Проверка предусловий и постусловий	281
Безопасные библиотеки для работы с целыми числами	283
Обнаружение переполнения	284
Генерируемые компилятором проверки времени выполнения	285
Операции с верифицируемым диапазоном	286
Модель AIR	287
Тестирование и анализ	288
5.7. Резюме	291
<b>Глава 6. Форматированный вывод</b>	<b>293</b>
6.1. Вариативные функции	294
6.2. Функции форматированного вывода	297
Строки формата	298
GCC	301
Visual C++	301
6.3. Использование уязвимостей функций форматированного вывода	302
Переполнение буфера	302
Выходные потоки	303
Аварийное завершение программы	303
Просмотр содержимого стека	304
Просмотр содержимого памяти	306
Перезапись памяти	307
Интернационализация	312
Уязвимости строк формата из широких символов	312
6.4. Рандомизация стека	312
Преодоление рандомизации стека	313
Запись адресов в двух словах	314
Непосредственный доступ к аргументу	315
6.5. Стратегии противодействия	317
Исключение пользовательского ввода из строк формата	317
Динамическое применение статического содержимого	317
Ограничение количества записанных байтов	318
Интерфейсы с проверкой границ из приложения К стандарта C11	319

iostream и stdio	320
Тестирование	321
Проверки компилятора	321
Статический анализ загрязненности	322
Изменение реализаций вариативных функций	322
Exec Shield	324
FormatGuard	324
Статический бинарный анализ	325
6.6. Известные уязвимости	326
Washington University FTP Daemon	326
CDE ToolTalk	326
Ettercap Version NG-0.7.2	327
6.7. Резюме	327
6.8. Дополнительная литература	328
<b>Глава 7. Параллельное выполнение</b>	<b>329</b>
7.1. Многопоточность	329
7.2. Параллельность	331
Параллелизм данных	332
Параллелизм задач	334
7.3. Производительность	334
Закон Амдаля	335
7.4. Распространенные ошибки	337
Состояния гонки	337
Поврежденные значения	338
Объекты volatile	339
7.5. Стратегии противодействия	341
Модель памяти	342
Примитивы синхронизации	344
Анализ роли потока (исследования)	352
Неизменяемые структуры данных	354
Свойства параллельного кода	355
7.6. Ловушки при контрмерах	356
Клинч	357
Преждевременное освобождение блокировки	361
Конкуренция	363
Проблема ABA	363
7.7. Известные уязвимости	368
DoS-атаки в многоядерных DRAM-системах	368
Уязвимости параллельности в оболочках системных вызовов	369
7.8. Резюме	370
<b>Глава 8. Файловый ввод-вывод</b>	<b>373</b>
8.1. Основы файлового ввода-вывода	373
Файловые системы	374
Специальные файлы	376

8.2. Интерфейсы файлового ввода-вывода	376
Потоки данных	377
Открытие и закрытие файлов	378
POSIX	379
Файловый ввод-вывод в C++	380
8.3. Управление доступом	381
Права доступа в UNIX	382
Привилегии процесса	384
Изменение привилегий	386
Управление привилегиями	389
Управление правами доступа	395
8.4. Идентификация файла	398
Обход каталогов	398
Ошибки эквивалентности	401
Символические ссылки	402
Канонизация	404
Жесткие ссылки	407
Файлы устройств	409
Атрибуты файла	412
8.5. Состояния гонки	414
Время проверки, время использования	415
Создание без замены	416
Эксклюзивный доступ	419
Совместно используемые каталоги	421
8.6. Стратегии противодействия	424
Закрытие окна гонки	424
Устранение объекта гонки	428
Управляемый доступ к объекту гонки	430
Инструменты обнаружения гонки	432
8.7. Резюме	433
<b>Глава 9. Рекомендованные практики</b>	<b>435</b>
9.1. Жизненный цикл разработки безопасного программного обеспечения	436
TSP-Secure	438
Планирование и отслеживание	439
Управление качеством	440
9.2. Обучение	441
9.3. Требования	443
Стандарты безопасного кодирования	443
Инжиниринг требований безопасности	444
Сценарии использования и злоупотребления	445
9.4. Проектирование	447
Принципы разработки безопасного программного обеспечения	449
Моделирование угроз	452
Анализ атак	453
Уязвимости в имеющемся коде	454

Безопасные оболочки	455
Проверка входных данных	456
Границы доверия	457
Черные списки	460
Белые списки	460
Тестирование	461
9.5. Реализация	461
Обеспечение безопасности компилятором	461
Модель AIR	463
Надежный и безопасный C/C++	463
Статический анализ	465
Source Code Analysis Laboratory (SCALe)	467
Глубокая защита	468
9.6. Верификация	469
Статический анализ	469
Проникающее тестирование	470
Нечеткое тестирование	470
Аудит кода	472
Руководства и контрольные списки разработчиков	472
Независимый обзор безопасности	473
Обзор области атак	473
9.7. Резюме	474
9.8. Дополнительная литература	474
<b>Список литературы</b>	<b>475</b>
<b>Аббревиатуры</b>	<b>488</b>
<b>Предметный указатель</b>	<b>494</b>