

19 СМЕРТНЫХ ГРЕХОВ, УГРОЖАЮЩИХ БЕЗОПАСНОСТИ ПРОГРАММ

Как не допустить типичных ошибок

Сделайте свои программы безопасными, исключив с самого начала причины возможных уязвимостей. Эта книга необходима всем разработчикам программного обеспечения, независимо от платформы, языка или вида приложений. В ней рассмотрены 19 грехов, угрожающих безопасности программ, и показано как от них избавиться. Авторы бестселлеров Майкл Ховард и Дэвид Лебланк, обучающие программистов в Microsoft, как писать безопасный код, объединили усилия с Джоном Виега, человеком, который сформулировал 19 смертных грехов программиста, и решили написать это руководство. На различных примерах продемонстрированы как сами ошибки, так и способы их исправления и защиты от них.

Если вы - программист, то вам просто необходимо прочесть эту книгу.

ИЗБАВЬТЕСЬ ОТ СЛЕДУЮЩИХ ИЗЪЯНОВ В СВОИХ ПРОГРАММАХ

- > Переполнение буфера
- > Ошибки, связанные с форматной строкой
- > Переполнение целых чисел
- > Внедрение SQL-команд
- > Внедрение команд
- > Пренебрежение обработкой ошибок
- > Кросс-сайтовые сценарии
- > Пренебрежение защитой сетевого трафика
- > Применение загадочных URL и скрытых полей форм
- > Неправильное применение SSL и TLS
- > Использование слабых систем на основе паролей
- > Пренебрежение безопасным хранением и защитой данных
- > Утечка информации
- > Некорректный доступ к файлам
- > Излишнее доверие к системе разрешения сетевых имен
- > Гонки
- > Неаутентифицированный обмен ключами
- > Использование случайных числа некриптографического качества
- > Неудобный интерфейс

Майкл Ховард является архитектором жизненного цикла разработки безопасного программного обеспечения в компании Microsoft и одним из авторов документа «Processes to Produce Secure Software». Он работает менеджером по безопасности программ в Microsoft.

Дэвид Лебланк, доктор философии, в настоящее время работает главным архитектором программ в компании Webroot Software. До этого он занимал

должность архитектора подсистемы безопасности в подразделении Microsoft, занимающемся разработкой Microsoft Office.

Джон Виега – технический директор компании Secure Software (www.securesoftware.com). Он первым дал описание 19 смертных грехов, угрожающих безопасности программ. Джон – автор многих книг по безопасности программного обеспечения.



19 СМЕРТНЫХ ГРЕХОВ



УГРОЖАЮЩИХ БЕЗОПАСНОСТИ ПРОГРАММ



19 СМЕРТНЫХ ГРЕХОВ

Windows
UNIX
Linux
Mac OS X
C
C++
C#
Java
PHP
Perl
Visual Basic
Web
Database



УГРОЖАЮЩИХ БЕЗОПАСНОСТИ ПРОГРАММ

Как не допустить
типичных ошибок

Майкл Ховард, Дэвид Лебланк, Джон Виега

19
СМЕРТНЫХ
ГРЕХОВ,

УГРОЖАЮЩИХ
БЕЗОПАСНОСТИ
ПРОГРАММ

Как не допустить типичных ошибок

МАЙКЛ ХОВАРД
ДЭВИД ЛЕБЛАНК
ДЖОН ВИЕГА

Москва

УДК 004.4
ББК 32.973.26-018.2
М97

Ховард М., Лебланк Д., Виера Д.

X68 19 смертных грехов, угрожающих безопасности программ.
Как не допустить типичных ошибок . – М.: ДМК Пресс. – 288 с.: ил.

ISBN 5-9706-0027-X

Эта книга необходима всем разработчикам программного обеспечения, независимо от платформы, языка или вида приложений. В ней рассмотрены 19 грехов, угрожающих безопасности программ, и показано как от них избавиться. Рассмотрены уязвимости на языках C/C++, C#, Java, Visual Basic, Visual Basic .NET, Perl, Python в операционных системах Windows, Unix, Linux, Mac OS, Novell Netware. Авторы издания, Майкл Ховард и Дэвид Лебланк, обучают программистов как писать безопасный код в компании Microsoft. На различных примерах продемонстрированы как сами ошибки, так и способы их исправления и защиты от них. Если вы – программист, то вам просто необходимо прочесть эту книгу.

УДК 004.4
ББК 32.973.26-018.2

Original English language edition published by McGraw-Hill Companies. Copyright © by McGraw-Hill Companies. All rights reserved.

Все права защищены. Любая часть этой книги не может быть воспроизведена в какой бы то ни было форме и какими бы то ни было средствами без письменного разрешения владельцев авторских прав.

Материал, изложенный в данной книге, многократно проверен. Но поскольку вероятность технических ошибок все равно существует, издательство не может гарантировать абсолютную точность и правильность приводимых сведений. В связи с этим издательство не несет ответственности за возможные ошибки, связанные с использованием книги.

ISBN 0-07-226085-8 (англ.) Copyright © by McGraw-Hill Companies.
ISBN 5-9706-0027-X © Перевод на русский язык, оформление, издание,
Издательский Дом ДМК-пресс

Содержание

Об авторах	18
О научных редакторах	19
Предисловие	20
Благодарности	22
Введение	23
Структура книги	24
Кому предназначена эта книга	25
Какие главы следует прочитать	25
Грех 1. Переполнение буфера	26
В чем состоит грех	26
Подверженные греху языки	27
Как происходит грехопадение	27
Греховность C/C++	31
Родственные грехи	33
Где искать ошибку	33
Выявление ошибки на этапе анализа кода	33
Тестирование	34
Примеры из реальной жизни	35
CVE-1999-0042	35
CVE-2000-0389 – CVE-2000-0392	35
CVE-2002-0842, CVE-2003-0095, CAN-2003-0096	35
CAN-2003-0352	36
Искупление греха	37
Замена опасных функций работы со строками	37
Следите за выделениями памяти	37
Проверьте циклы и доступ к массивам	37
Пользуйтесь строками в стиле C++, а не C	37
Пользуйтесь STL-контейнерами вместо статических массивов	38

Пользуйтесь инструментами анализа	38
Дополнительные защитные меры	38
Защита стека	39
Запрет исполнения в стеке и куче	39
Другие ресурсы	39
Резюме	40
Грех 2. Ошибки, связанные с форматной строкой	42
В чем состоит грех	42
Подверженные греху языки	42
Как происходит грехопадение	43
Греховность C/C++	45
Родственные грехи	45
Где искать ошибку	46
Выявление ошибки на этапе анализа кода	46
Тестирование	46
Примеры из реальной жизни	47
CVE-2000-0573	47
CVE-2000-0844	47
Искупление греха	47
Искупление греха в C/C++	48
Дополнительные защитные меры	48
Другие ресурсы	48
Резюме	48
Грех 3. Переполнение целых чисел	49
В чем состоит грех	49
Подверженные греху языки	49
Как происходит грехопадение	49
Греховность C и C++	50
Поразрядные операции	55
Греховность C#	55
Греховность Visual Basic и Visual Basic .NET	56
Греховность Java	57
Греховность Perl	58
Где искать ошибку	59
Выявление ошибки на этапе анализа кода	59
C/C++	59
C#	61
Java	62
Visual Basic и Visual Basic .NET	62

Perl	62
Тестирование	62
Примеры из реальной жизни	62
Ошибка в интерпретаторе Windows Script позволяет выполнить произвольный код	63
Переполнение целого в конструкторе объекта SOAPParameter	63
Переполнение кучи в HTR-документе, передаваемом поблочно, может скомпрометировать Web-сервер	63
Искупление греха	64
Дополнительные защитные меры	66
Другие ресурсы	66
Резюме	66
Не рекомендуется	66
Грех 4. Внедрение SQL-команд	67
В чем состоит грех	67
Подверженные греху языки	67
Как происходит грехопадение	68
Греховность C#	68
Греховность PHP	69
Греховность Perl/CGI	69
Греховность Java	70
Греховность SQL	71
Родственные грехи	72
Где искать ошибку	72
Выявление ошибки на этапе анализа кода	72
Тестирование	73
Примеры из реальной жизни	75
CAN-2004-0348	75
CAN-2002-0554	75
Искупление греха	75
Проверяйте все входные данные	76
Никогда не применяйте конкатенацию для построения SQL-предложений	76
Дополнительные защитные меры	79
Другие ресурсы	79
Резюме	80
Грех 5. Внедрение команд	82
В чем состоит грех	82

Подверженные греху языки	82
Как происходит грехопадение	82
Родственные грехи	84
Где искать ошибку	84
Выявление ошибки на этапе анализа кода	84
Тестирование	86
Примеры из реальной жизни	86
CAN-2001-1187	86
CAN-2002-0652	87
Искупление греха	87
Контроль данных	87
Если проверка не проходит	90
Дополнительные защитные меры	90
Другие ресурсы	91
Резюме	91
Грех 6. Пренебрежение обработкой ошибок	92
В чем состоит грех	92
Подверженные греху языки	92
Как происходит грехопадение	92
Раскрытие излишней информации	92
Игнорирование ошибок	93
Неправильная интерпретация ошибок	93
Бесполезные возвращаемые значения	94
Обработка не тех исключений, что нужно	94
Обработка всех исключений	94
Греховность C/C++	94
Греховность C/C++ в Windows	95
Греховность C++	96
Греховность C#, VB.NET и Java	96
Родственные грехи	97
Где искать ошибку	97
Выявление ошибки на этапе анализа кода	97
Тестирование	97
Примеры из реальной жизни	98
CAN-2004-0077 do_mremap в ядре Linux	98
Искупление греха	98
Искупление греха в C/C++	98
Искупление греха в C#, VB.NET и Java	99
Другие ресурсы	99
Резюме	100

Грех 7. Кросс-сайтовые сценарии	101
В чем состоит грех	101
Подверженные греху языки	101
Как происходит грехопадение	101
Греховное ISAPI-расширение или фильтр на C/C++	102
Греховность ASP	103
Греховность форм ASP.NET	103
Греховность JSP	103
Греховность PHP	103
Греховность Perl-модуля CGI.pm	103
Греховность mod-perl	104
Где искать ошибку	104
Выявление ошибки на этапе анализа кода	104
Тестирование	105
Примеры из реальной жизни	106
Уязвимость IBM Lotus Domino для атаки с кросс-сайтовым сценарием и внедрением HTML	106
Ошибка при контроле входных данных в сценарии isqlplus, входящем в состав Oracle HTTP Server, позволяет удаленному пользователю провести атаку с кросс-сайтовым сценарием	106
CVE-2002-0840	107
Искупление греха	107
Искупление греха в ISAPI-расширениях и фильтрах на C/C++	107
Искупление греха в ASP	108
Искупление греха в ASP.NET	108
Искупление греха в JSP	108
Искупление греха в PHP	110
Искупление греха в Perl/CGI	110
Искупление греха в mod-perl	111
Замечание по поводу HTML-кодирования	111
Дополнительные защитные меры	112
Другие ресурсы	112
Резюме	113
Грех 8. Пренебрежение защитой сетевого трафика	114
В чем состоит грех	114
Подверженные греху языки	114
Как происходит грехопадение	115
Родственные грехи	117

Где искать ошибку	117
Выявление ошибки на этапе анализа кода	118
Тестирование	121
Примеры из реальной жизни	121
TCP/IP	121
Протоколы электронной почты	122
Протокол E*Trade	122
Искупление греха	122
Рекомендации низкого уровня	123
Дополнительные защитные меры	126
Другие ресурсы	126
Резюме	126

Грех 9. Применение загадочных URL и скрытых полей форм	128
В чем состоит грех	128
Подверженные греху языки	128
Как происходит грехопадение	128
Загадочные URL	128
Скрытые поля формы	129
Родственные грехи	129
Где искать ошибку	130
Выявление ошибки на этапе анализа кода	130
Тестирование	131
Примеры из реальной жизни	131
CAN-2000-1001	132
Модификация скрытого поля формы в программе MaxWebPortal	132
Искупление греха	132
Противник просматривает данные	132
Противник воспроизводит данные	133
Противник предсказывает данные	135
Противник изменяет данные	136
Дополнительные защитные меры	137
Другие ресурсы	137
Резюме	137

Грех 10. Неправильное применение SSL и TLS	138
В чем состоит грех	138
Подверженные греху языки	138
Как происходит грехопадение	139