ОПЫТ ВЫЯВЛЕНИЯ УЯЗВИМОСТЕЙ В ЗАРУБЕЖНЫХ ПРОГРАММНЫХ ПРОДУКТАХ

***Марков Алексей Сергеевич**, кандидат технических наук, старший научный сотрудник, CISSP*
***Цирлов Валентин Леонидович**, кандидат технических наук*

Рассмотрены вопросы безопасности импортных программах средств. Приведена статистика по выявленным уязвимостям в импортных программных продуктах. Рассмотрены общие методы выявления уязвимостей и дефектов безопасности программных средств.

***Ключевые слова**: безопасность программ, дефекты безопасности, уязвимости, сертификация средств защиты информации*

EXPERIENCE IN IDENTIFYING VULNERABILITIES IN SOFTWARE

***Alexey Markov** , Ph.D., Associate Professor, CISSP*
***Valentin Tsirlov**, Ph.D*

Questions of security software are considered. The statistics on the identified vulnerabilities in the import software are submitted. The general methods for identifying security software vulnerabilities and defects are discussed.

***Keywords**: security software, security defects, vulnerability, information security certification, information security conformity assessment*

Введение

В рамках обсуждения проблем кибербезопасности остро стоит вопрос о наличии разного рода уязвимостей в программных продуктах импортного производства. В открытых источниках периодически обсуждаются скандалы с обнаружением программных закладок в продукции мировых разработчиков информационных технологий. Одним из способов противодействия подобным угрозам является проверка безопасности программного кода в процессе сертификационных испытаний и тематических исследований по требованиям безопасности. В данной статье приведены примеры выявленных уязвимостей исходя из опыта работы испытательной лаборатории.

Виды сертификационных испытаний

В Российской Федерации основным легитимным способом выявления уязвимостей ПО является *обязательная сертификация средств защиты информации по требованиям безопасности информации* (по линии Минобороны России, ФСБ России и ФСТЭК России) [1]. Это связано с тем, что при сертификации официально предоставляются необходимые спецификации, имеется обратная связь с разработчиком, а в ряде случаев предоставляется исходный программный код, компоновочная среда и т.д.

Сертификационные испытания и тематические исследования, регламентированные современной нормативной базой, проводятся путем:

- функционального тестирования на соответствие нормативным и методическим документам или документации (ТУ, формуляр, задание по безопасности);
- структурного декомпозиционного анализа программного обеспечения на отсутствие недекларированных возможностей [2].

Особенностями указанных подходов является следующее:

1. Функциональное тестирование программ касается проверки задекларированных детерминированных механизмов безопасности, т.е. проверяется факт их работы, не касаясь глубокого анализа защищенности. Однако, используя личный опыт, квалифицированные эксперты способны построить тесты, позволяющие выявлять некоторые специфические ошибки безопасности проектирования, реализации, конфигураций, прототипов, интерфейсов и т.д.

2. При структурном анализе импортной продукции (если он предусмотрен) проводится, главным образом, проверка полноты/избыточности кода. При проверке программных средств защиты информации, отнесенной к гостайне, также должен проводиться еще статический и динамический анализ, который заключается в выполнении декомпозиции программной системы (формировании и контроле условной части маршрутов).

Однако нормативная база не ограничивает экспертов в использовании дополнительных методов и приемов проверки кода, например: инспекции кода, использовании статических анализаторов, изучении бюллетеней безопасности, организации фаззинг - и стресс-тестирования и др.

Виды тестирования безопасности кода

Опираясь на методологию риск-менеджмента, при тестировании безопасности программного кода следует сформулировать вертикаль факторов ИБ:

{ДЕФЕКТЫ} -> {УЯЗВИМОСТИ} -> {УГРОЗЫ} -> {РИСКИ}

В названном перечне, с точки зрения анализа кода, первичными являются именно *дефекты безопасности*, которые представляют собой потенциальные уязвимости, влияющие на целостность, доступность, конфиденциальность ресурсов. Дефекты, которые локализованы, описаны, эксплуатируемы, идентифицируются как *уязвимости*. Как правило, дефекты выявляются на этапе аудита безопасности кода, а уязвимости выявляются при сканировании информационной системы (сопоставлении идентифицируемых программ базе описаний уязвимостей или проверке кода программы на наличие сигнатуры уязвимости).

Роль и место указанных факторов в рамках модели управления безопасностью программного обеспечения (ПО) представлены в табл. 1.

В настоящее время методы и технологии выявления уязвимостей *не носят универсальный характер* и ориентированы на определенные классы уязвимостей и их причин (дефектов).

На практике выделяют три условных класса дефектов и уязвимостей:

1. «Некорректности программирования», классифицируемые как нефункциональные ошибки, сделанные при кодировании и влияющие на конфиденциальность, целостность, доступность ресурсов. Теоретически, такие дефекты могут быть внесены умышлено.

При тестировании обычно полагается, что такие дефекты имеют стохастический характер, т.е. для выявления применяются методы функционального тестирования (обычно, фаззинг-тестирование). К примеру, по заявлению разработчиков, бета-версия Windows 8 прошла 1 миллиард запусков.

В настоящее время развивается направление прикладной верификации кода, позволяющей в рамках статического анализа найти «некорректности программирования»: переполнение буфера, избыточные переменные и объекты и др.

2. Дефекты, идентифицируемые как преднамеренные. Так как такие дефекты связаны с редкими входными данными, то в реальное время их можно выявить только ручными экспертными и полуавтоматизируемыми сигнатурными (эвристическими) методами [3].

Таблица 1

Управление безопасностью программ

Фактор	Управление		
	Анализ/тестирование	Контроль	Контрмеры
Дефекты	Выявление, локализация	Инспекционный контроль ПО	Безопасное программирование
Уязвимости	идентификация, сканирование	Периодическое сканирование, контроль целостности, контроль источников происхождения компонентов и др.	Исправления
Угрозы	Формирование модели угроз	Мониторинг угроз	Обновления, блокировка, фильтрация и др.
Риски	Оценка риска	Оценка остаточного риска	Обработка риска

3. Ранее обнаруженные (известные) уязвимости, которые выявляются методами сканирования и экспертными методами, включающими также сбор и анализ бюллетеней, прототипов и т.д.

При отсутствии исходных данных применяются подходы реверс-инжиниринга и функциональные методы (по принципу «черного ящика»). Реверс-инжиниринг может проводиться путем:

- ретрансляции/дизассемблирования, прогона в отладочном режиме - для машинных и процедурных языков;

- высококачественной декомпиляцией - для языков с промежуточным кодом [4].

Надо понимать, что все методы имеют ограничения по использованию:

- функциональные методы ограничены проклятием размерности входных данных, неэффективны при выявлении программных закладок и пригодны для небольших продуктов ;

- структурные статические методы, кроме наличия исходных текстов, имеют ограничения на выявление дефектов, связанных с динамикой программы (циклами и т.д.);

- дизассемблирование - реально провести для небольших незащищенных программ;

- ручные экспертные методы предъявляют высокие требования к опыту и знаниям тестировщиков.

Примеры отдельных техник тестирования представлены в табл.2 [2].

Важным моментом при выявлении уязвимостей является сочетание методов тестирования и методов *мониторинга* информационной безопасности (ИБ), включая реверсинг трафика и контроль событий ИБ.

Таким образом, использование различных техник проверки кода в рамках общей организации сертификации импортной программной продукции позволяет выявить ряд дефектов и уязвимостей, статистика по которым представлена ниже.

Статистика выявления уязвимостей

В процессе сертификации ПО испытательной лабораторией было выявлено несколько десятков дефектов ПО, идентифицированных как критические уязвимости, и более тысячи дефектов безопасности, которые идентифицировать как

Таблица 2.

Примеры техник тестирования средств защиты информации

Метод тестирования	Основные выявляемые дефекты и уязвимости
Функциональное тестирование	Дефекты реализации функций и ошибки документации
Фаззинг-тестирование	Дефекты реализации интерфейсов данных
Граничное тестирование	Ошибки граничных условий
Нагрузочное тестирование	Ошибки производительности
Стресс-тестирование (а разве это не одно и тоже, что нагрузочное? Просто разные слова)	Отказ в обслуживании
Профилирование	Недостатки оптимизации кода
Статический семантический анализ (прикладная верификация)	Некорректности кодирования
Статический сигнатурный анализ	Заданные потенциально опасные фрагменты
Статический анализ отсутствия недекларируемых возможностей (НДВ)	«Мертвый код»
Динамический анализ отсутствия НДВ	«Мертвый код»
Мониторинг операционных процессов	Нарушения целостности процессов и ресурсов
Тестирование конфигураций	Ошибки администрирования
Сканирование уязвимостей	Известные опубликованные уязвимости
Тест на проникновение	Известные уязвимости, ошибки конфигурирования
Регрессионное тестирование	Повторные ошибки прошлых версий

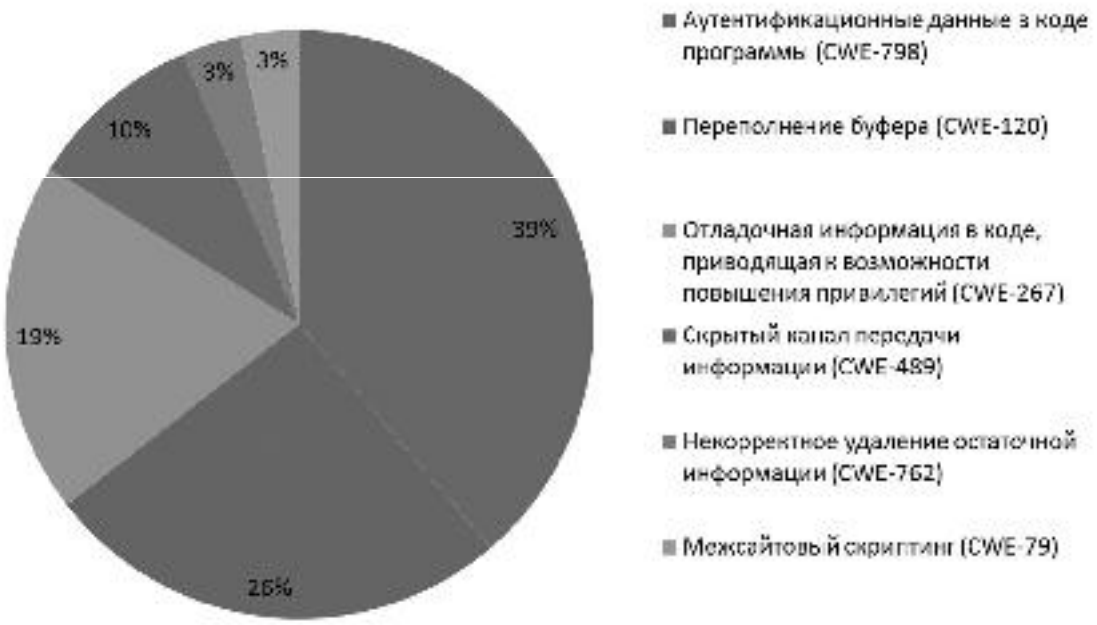

Рис. 1. *Статистика по типам уязвимостям*

преднамеренные не удалось. Под проверки подпала продукция 15 зарубежных производителей из 7 иностранных держав.

Статистика по типам уязвимостей

Испытания показали, что в ПО в явном виде встречаются программные закладки, маскируемые под отладочные средства (встроенные учетные записи и мастер-пароли, а также средства удаленного управления). Около 70% выявленных уязвимостей являются именно такими. В то же время зафиксирован ряд дефектов, которые трудно идентифицировать как преднамеренные, однако их можно эксплуатировать при проведении компьютерных атак, например, межсайтовый скриптинг (Cross-Site Scripting - XSS).

Статистика уязвимостей по типам программ

Из зарубежной продукции в рамках сертификации были проверены операционные системы, антивирусные решения, системы обнаружения вторжений, системы хранения информации и автоматизации предприятий, сетевые устройства. Статистика соответствует общеми-

Рис. 2. *Статистика уязвимостей по типам ПО*

Рис. 3. *Статистика по методам выявления уязвимостей*

ровой – большинство уязвимостей обнаружено в прикладных системах.

Следует отметить, что во всех образцах телекоммуникационного оборудования были обнаружены встроенные учетные записи (CWE-798).

Статистика по методам тестирования

Подавляющее большинство уязвимостей было выявлено методами статического эвристического (сигнатурного) анализа. Для сравнения, следует отметить, что в практике проверки российского ПО доля уязвимостей (главным образом ошибок кодирования), выявленных функциональными методами, существенно выше (до 30%), чем для зарубежного. Это легко можно объяснить наличием сертифицированных систем менеджмента информационной безопасности (СМИБ) на зарубежных предприятиях.

Статистика дефектов в открытом коде

Следует указать, что современные программные комплексы включают модули программ с открытым кодом. Исследование показало, что такие программы тоже включают уязвимости. Ниже представленная статистика демонстрирует наличие уязвимостей в открытом коде (рис.4).

Краткие выводы по сертификационной статистике

Можно сделать краткие выводы:

- большинство импортных программных продуктов имели программные закладки аутентификационного (расшифруйте полнее) характера и др.;
- подавляющее большинство уязвимостей было выявлено только в случае предоставления исходных кодов;

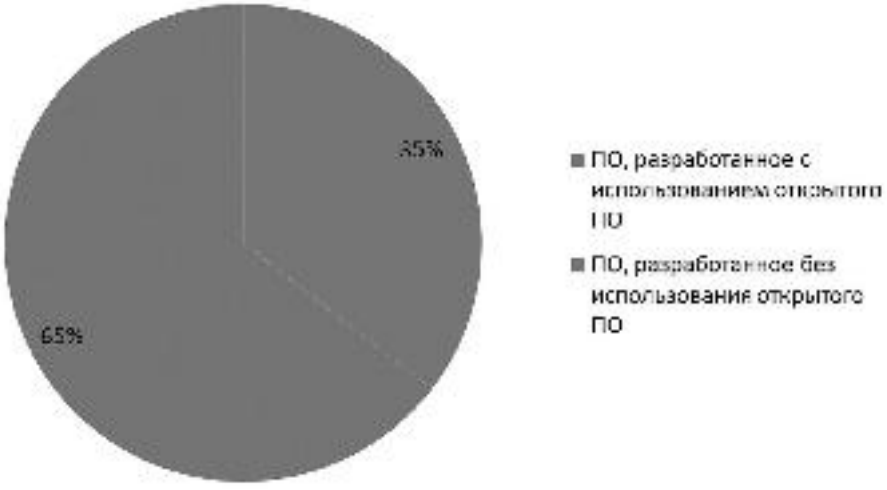

Рис. 4. *Продукты с открытым кодом тоже содержат уязвимости*

- большинство уязвимостей зафиксировано на уровне прикладных приложений (а не средств защиты информации);

- количество найденных уязвимостей, не идентифицированных как преднамеренные, зависит от существующей в организации системы менеджмента информационной безопасности (жизненного цикла безопасного производства программ).

Состояние проблемы в зарубежных странах

Так как наличие программных уязвимостей является основой реализации современных кибератак, то интересно познакомиться с зарубежным опытом в области кибербезопасности. Так, например, США в настоящее время активизируется внимание к активным методам информационного противоборства. Можно отметить ряд тенденций:

1. В области ИБ в США очевиден упор на выявление и эксплуатацию уязвимостей. АНБ в настоящее время демонстрирует привлечение «хакерских» технологий», например, ведет несколько десятков крупномасштабных проектов, включая создание датацентров АНБ, центров обучения по кибербезопасности, привлечение хакеров на работу в АНБ [5-7].

2. США традиционно ведет политику «черных списков» для зарубежных разработчиков ПО. В настоящее время в США озабочены противодействием китайским технологиям в военном секторе [8]. В стране имеется демонстративная система поставщиков в DoD.

3. Программное обеспечение в государственных структурах подлежит в обязательном порядке проверкам исходного кода, по результатам которого предусмотрено внедрение методов противодействия недоверенному ПО [9-11]. В ряде других сфер (например, во всех платежных системах) аудит безопасности ПО «добровольно принудительный».

При сертификации критических систем в обязательном порядке проводится тестирование на проникновение (включая аудит безопасности кода). При сертификации средств защиты введено обязательное тестирование на проникновение, а также проведена трансформация методологии испытаний от показателей «качества» к показателям «безопасности». В последнем случае, можно отметить консолидацию европейских стран по изменению процедуры сертификационных испытаний. Например, сертификация в Франции – CSPN, которая заметно проще в вопросах оценки доверия, но отличительной особенностью которой является обязательное тестирование на проникновение [12].

Заключение

Можно выделить стратегические задачи, касающиеся снижения угроз, связанных с наличием уязвимостей в импортном ПО:

- повышение контроля защищенности систем, путем внедрения технологий анализа защищённости, в первую очередь, тестирования на проникновение;

- повышение контроля безопасности программной продукции, путем внедрения технологий аудита безопасности кода;

- разработка технологий функционирования систем в условиях наличия угроз, связанных с использованием недоверенной программной продукции.

Решение указанных задач затрагивает вопросы совершенствования нормативно-правовой базы, аккредитации и обучения, развития информационно-технического обеспечения испытаний и др.

Литература

1. Марков А.С., Цирлов В.Л. Сертификация программ: мифы и реальность // Открытые системы. СУБД. 2011. № 6. С. 26-29.
2. Марков А.С., Цирлов В.Л., Барабанов А.В. Методы оценки несоответствия средств защиты информации. М.: Радио и связь, 2012. 192 с.
3. Марков А.С., Фадин А.А. Статический сигнатурный анализ безопасности программ // Программная инженерия и информационная безопасность. 2013. №1(1). С.50-56.
4. Барабанов А.В., Марков А.С., Фадин А.А. Сертификация программ без исходных текстов // Открытые системы. СУБД. 2011. № 4. С. 38-41.
5. Department of Defense Fiscal Year (FY) 2014 President's Budget Submission. US Department of the Army, 2013. 679 p.

6. Department of Defense Fiscal Year (FY) 2014 President's Budget Submission. DARPA. 2013. 318 p.
7. Cowley S. NSA wants to hire hackers // CNN Money. 2012/07/27, pp 1-1.
8. Investigative Report on the U.S. National Security Issues Posed by Chinese Telecommunications Companies Huawei and ZTE, U.S. House of Representatives, 2012 60 p.
9. Improvements in assurance of computer software procured by the Department of Defense // National Defense Authorization Act for Fiscal Year 2013. Sec. 933. P.253, 254.
10. FIPS PUB 200. Minimum Security Requirements for Federal Information and Information Systems. NIST 2006. 17 p.
11. NIST Special Publication 800-53. Security and Privacy Controls for Federal Information Systems and Organizations. 2013. Revision 4. P. 457.
12. First level security certification for information technologies (CSPN) // ANSSI-CSPN-CER-P.01.1EN. 2011. 11 p.

References

1. Markov A.S., Tsirlov V.L. Sertifikatsiya programm: mify i realnost, (Certification programs: myths and reality), Otkrytyye sistemy. SUBD, 2011, No 6, pp. 26-29.
2. Markov A.S., Tsirlov V.L., Barabanov A.V. Metody otsenki nesootvetstviya sredstv zashchity informatsii, Moscow, Radio i Svyaz, 2012, pp. 1-192.
3. Markov A.S., Fadin A.A. Staticheskiy signaturnyy analiz bezopasnosti program, (Static signature-based security analysis program), Programmnaya inzheneriya i informatsionnaya bezopasnost, 2013, No 1(1), pp. 50-56.
4. Barabanov A.V., Markov A.S., Fadin A.A. Sertifikatsiya programm bez iskhodnykh tekstov, (Certification programs without source), Otkrytyye sistemy. SUBD, 2011, No 4, pp. 38-41.
5. Department of Defense Fiscal Year (FY) 2014 President's Budget Submission. US Department of the Army, 2013. 679 p.
6. Department of Defense Fiscal Year (FY) 2014 President's Budget Submission. DARPA. 2013. 318 p.
7. Cowley S. NSA wants to hire hackers // CNN Money. 2012/07/27, pp 1-1.
8. Investigative Report on the U.S. National Security Issues Posed by Chinese Telecommunications Companies Huawei and ZTE, U.S. House of Representatives, 2012 60 p.
9. Improvements in assurance of computer software procured by the Department of Defense // National Defense Authorization Act for Fiscal Year 2013. Sec. 933. P.253, 254.
10. FIPS PUB 200. Minimum Security Requirements for Federal Information and Information Systems. NIST 2006. 17 p.
11. NIST Special Publication 800-53. Security and Privacy Controls for Federal Information Systems and Organizations. 2013. Revision 4. P. 457.
12. First level security certification for information technologies (CSPN) // ANSSI-CSPN-CER-P.01.1EN. 2011. 11 p.