УДК 681.3.07

И.В. Жидков, О.Н. Федорец

Проблема создания безопасного программного обеспечения и предложения по ее решению

Проведен анализ состояния вопроса разработки программного обеспечения (ПО).
Разработаны предложения по повышению безопасности ПО.

Роль безопасности программного обеспечения с каждым годом неуклонно растет, в то же время растет число инцидентов, связанных с вопросами его безопасности.

Многие уязвимые места появляются вследствие ошибок, которые непреднамеренно возникают при проектировании и разработке программного обеспечения. Согласно анализу, выполненному специалистами CERT/CC, источниками более 90% уязвимых мест являются известные типы дефектов. Под дефектом будем понимать все то, что обусловливает необходимость каких-либо изменений продукта (вне зависимости от того, связано ли это с несоответствием требованиям, неудачными конструкторскими решениями, недостаточным уровнем безопасности и удобства использования или с ошибками кодирования).

При использовании сегодняшних технологий разработки в лучшем случае на каждую тысячу строк нового или измененного кода приходится, как правило, от 1 до 10 дефектов. Следовательно, в типичных современных системах, содержащих миллионы строк кода, имеются тысячи дефектов. Естественно, такое программное обеспечение не может гарантировать безопасность. Для создания действительно безопасного программного обеспечения организациям необходимо сократить на один-два порядка число дефектов в спецификациях, ошибок при проектировании и реализации.

С целью частичного решения данной проблемы в испытательной лаборатории ФГУ «3 ЦНИИ Минобороны России» разработано программное средство поиска уязвимых функций в исходных текстах программ (ПУФИТП), предназначенное для анализа исходных текстов программ, написанных на языках С и С++ на предмет наличия уязвимых функций, которые могут привести к нарушению безопасности информации [1].

Программное средство ПУФИТП обеспечивает выполнение следующих функций:

– выявление уязвимых функций, которые могут привести к нарушению безопасности информации;
– систематизация выявленных уязвимых функций по степени риска;
– пополнение имеющейся базы данных уязвимых функций;
– протоколирование результатов в различных видах (по выбору пользователя).

На программное средство ПУФИТП получено свидетельство о регистрации программы для ЭВМ [2]. Оно активно используется экспертами испытательной лаборатории ФГУ «3 ЦНИИ Минобороны России» в процессе проведения сертификационных испытаний. По результатам его работы разработчикам программной продукции выдаются рекомендации по доработке программных изделий с целью повышения безопасности информации.

Кроме того, ПУФИТП может применяться разработчиками программного обеспечения на всех этапах разработки ПО с целью обеспечения безопасности программного обеспечения.

Анализ состояния вопроса разработки ПО выявил, что разработкой программного обеспечения занимается большое число различных организаций, в которых:

1) квалификация персонала различна;

2) процесс разработки программного обеспечения построен по-разному. В одних организациях разработкой ПО занимается только несколько человек без использования какого-либо управления проектами, в других разработкой ПО занимаются десятки и сотни человек с четким управлением и разделением труда, а также тщательным тестированием ПО на всех этапах разработки;

3) отсутствует контроль ПО на наличие уязвимостей как при разработке, так и на всех этапах испытаний.

Это результат отсутствия соответствующих единых обязательных требований со стороны министерств (ведомств). Все это свидетельствует о том, что разрабатываемое ПО является далеко не безопасным, несмотря на его многочисленные испытания.

В то же время подходы к повышению безопасности разрабатываемого ПО существуют. Сформулируем основные предложения по повышению безопасности ПО.

1. Необходимо определить основные процессы разработки ПО с малым числом дефектов. Чтобы начать двигаться в этом направлении, любая организация, по той или иной причине намеренная заняться разработкой безопасного ПО, должна использовать процесс, который вполне предсказуемо позволит получать программное обеспечение менее чем с 0,1 дефекта в спецификациях, архитектуре и реализации на каждую тысячу строк нового и измененного кода.

2. Необходимо сформировать центр управления рисками для принятия решений. Это потребует опыта работы со средствами обеспечения безопасности и знаний, охватывающих все аспекты создаваемой системы. А поскольку в данном случае нужны действительно обширные и глубокие знания, может понадобиться помощь экспертов в выявлении и управлении рисками.

3. Необходимо разрабатывать спецификации и архитектуру продукта с учетом вопросов безопасности. Проектирование критически важных для защиты аспектов должно быть сконцентрировано в

ограниченной части ПО. Архитектура должна быть как можно более простой — возможно, даже в ущерб эффективности. Следует ограничиться применением «безопасных» структур и функций, предпочтительно из числа поддающихся анализу. Архитектура строится исходя из того, что программа может подвергнуться разрушению и должна обеспечивать эшелонированную защиту и быть устойчивой к различным воздействиям.

4. Для обнаружения известных дефектов кода необходимо использовать специализированные средства статического и динамического анализа кода программ. Испытания должны предусматривать проведение серьезных атак, попытки вскрытия системы безопасности, поиск общих уязвимых мест. Выявление дефектов системы безопасности в ходе испытаний позволяет не только эффективно исправлять текущие ошибки, но и предотвращать их появление в будущих системах и проектах. Кроме того, аналогичные дефекты можно исключить из других программных продуктов.

5. Для выявления и ликвидации основных причин появления дефектов необходим постоянный контроль за разрабатываемыми продуктами и используемыми процессами. Проанализировав в процессе разработки и тестирования, а также после завершения реализации проекта характеристики и причины возникновения уязвимых мест, разработчики ПО должны извлечь из этого необходимые уроки.

6. При организации технической поддержки, исправлении ошибок и выпуске обновлений необходимо также использовать оптимальные методы и процессы, так как это играет не менее важную роль, чем разработка нового ПО. Особое внимание следует уделять процессам управления конфигурацией: изменения в уже существующие программы следует вносить только после определения четкой процедуры таких изменений и настройки конфигурации. Например, модификацию ПО возможно проводить путем замены отдельных модулей без изменения общей структуры.

7. Необходимо использовать только сертифицированные и выбранные в качестве единых инструментальных средств разработки программ.

8. Вместе со своими продуктами разработчики должны поставлять руководства по обеспечению безопасности. В них необходимо указывать все допущения, уровень безопасности используемых средств, а также давать рекомендации по настройке конфигурации программного обеспечения.

9. Необходимо создать типовую общую безопасную информационную базу алгоритмов, исходных текстов и программных средств, обеспечивающих информационную, технологическую и программную совместимость, на основе максимальной их унификации по всем компонентам и интерфейсам.

Кроме того, учитывая важность обеспечения безопасности информации критически важных систем, государство должно:

- стимулировать любого разработчика как можно быстрее внедрять процессы, которые позволяют создавать программное обеспечение, практически не имеющее дефектов как в спецификациях, так и в архитектуре и реализации;
- стимулировать организации, занимающиеся разработкой программного обеспечения, использовать передовой опыт создания безопасного ПО;
- разработать программу оценки и измерения основных параметров, которая позволит определять эффективность таких процессов и сертифицировать те из них, которые действительно подходят для выпуска безопасного программного обеспечения.

Таким образом, при создании безопасного программного обеспечения необходимо учитывать множество факторов, связанных с процессами его проектирования, обеспечением защиты и с организацией управления. Разработка должна начинаться с определения наилучших методов проектирования, дополняться хорошо зарекомендовавшими себя техническими подходами и подкрепляться управленческой практикой, способствующей удачному завершению процесса создания безопасного ПО, качество которого подтверждается обязательными сертификационными испытаниями с использованием различных программных средств автоматизации.

Работа выполнена при поддержке Гранта Президента Российской Федерации для государственной поддержки молодых российских ученых МК-2322.2007.10.

Литература

1. Жидков И.В., Львов В.М., Федорец О.Н. Инструментальные средства и технология тестирования и испытаний программного обеспечения // Информатизация и связь. – №2. –2007. – С. 49–54.

2. Зубарев И.В., Грибков В.В., Жидков И.В., Федорец О.Н. Свидетельство об официальной регистрации программы для ЭВМ №2008610253 от 8.11.2007.

Жидков Игорь Васильевич
ФГУ «3 ЦНИИ Минобороны России», к.т.н., начальник отдела
Эл. почта: igorzh@bk.ru.

Федорец Олег Николаевич
ФГУ «3 ЦНИИ Минобороны России», к.т.н., доцент, заместитель начальника отдела, начальник лаборатории
Эл. почта: burkos@mail.ru.

I.V. Zhidkov, O.N. Fedorets

Problem of creation of the safe software and the offers under its decision

The analysis of the question condition of the software development is lead. Offers on increase of the software safety are developed.