Введение

В этой книге даются конкретные рекомендации для программирующих на Java. Следуя этим рекомендациям, они смогут разрабатывать более надежные системы, устойчивые к нарушению защиты. Эти рекомендации охватывают широкий спектр программных продуктов, разрабатываемых на Java для таких устройств, как ПК, игровые приставки, мобильные телефоны, планшетные компьютеры, бытовая техника и автомобильная электроника.

Программирующие на любом языке должны придерживаться определенного ряда рекомендаций, касающихся управления структурами их программ, а самое главное — того, что указано в определении языка программирования. И это в равной степени относится к Java. Для разработки надежных и безопасных программ на Java программистам требуется дополнительная помощь, помимо того, что указано в спецификации языка программирования Java (JLS) [JLS 2013]. В состав Java входят языковые средства и прикладные программные интерфейсы (API), которые можно легко употребить неправильно, и поэтому требуются рекомендации, помогающие обойти скрытые препятствия на пути к созданию надежных программ на Java.

Для того чтобы программа была *надежной*, она должна работать во всех случаях и вопреки любым данным, которые могут быть введены. Любая нетривиальная программа не может избежать совершенно неожиданно возникающей ситуации в ходе ее выполнения или при вводе данных, в результате чего возникают ошибки. Когда же возникают ошибки, очень важно ограничить их воздействие на программу, и для этого лучше всего локализовать ошибку и обработать ее как можно скорее. Стремясь предусмотреть неожиданные ситуации, которые могут возникнуть при вводе данных или в ходе выполнения программы, одни программисты могут выгодно воспользоваться опытом других, приняв на вооружение безопасный стиль программирования.

Некоторые рекомендации по программированию носят стилистический характер, и все же они очень важны для обеспечения надежности и сопровождаемости кода. Для программирования на Java в компании Oracle был установлен ряд правил оформления кода [Conventions 2009] с целью помочь разработчикам выработать устоявшийся стиль программирования, и с тех пор эти правила были повсеместно приняты программирующими на Java.

Безопасное программирование на Java

Предлагаемые читателям рекомендации по программированию на Java изложены авторами, написавшими ранее книгу *The CERT® Oracle® Secure Coding Standard for Java™* [Long 2012]. Описываемые в этой книге нормы оформления кода устанавливают ряд правил для безопасного программирования на Java. Эти правила нацелены на то, чтобы искоренить из практики привычку к небезопасному программированию, зачастую приводящему к появлению уязвимостей, пригодных для незаконного использования. Стандарт безопасного программирования на Java устанавливает нормативные требования для систем программного обеспечения. Такие системы могут быть оценены на соответствие данному стандарту, например, в Лаборатории анализа исходного кода (Source Code Analysis Laboratory — SCALe) [Seacord 2012]. Тем не менее по-прежнему практикуются неправильные приемы программирования на Java, способные привести к появлению ненадежных или небезопасных программ, несмотря на то, что они исключены из стандарта безопасного программирования на Java. Поэтому цель данной книги — рассмотреть эти неправильные приемы программирования на Java и предостеречь от их применения на практике.

Несмотря на то что рассматриваемые здесь рекомендации не включены в материал книги *The CERT® Oracle® Secure Coding Standard for Java™*, это не преуменьшает их значение. Рекомендации должны быть исключены из стандарта на программирование, если по ним нельзя составить нормативные требования. Имеется немало причин, по которым нельзя составить нормативное требование, и самая распространенная среди них состоит в том, что всякое правило зависит от *намерения* программиста. Подобные правила не могут быть соблюдены автоматически, если только не определено намерение программиста. В таком случае правило может потребовать согласования кода и определенного намерения. При составлении нормативного требования необходимо также предписать, что нарушение этого требования означает дефект в коде. Рекомендации были исключены из стандарта на программирование, но вошли в материал этой книги, в тех случаях, когда их целесообразно соблюдать, хотя их несоблюдение не всегда приводит к ошибке. Такое различие проводится для того, чтобы систему программного обеспечения нельзя было считать не отвечающей стандарту в отсутствие конкретного дефекта в ее коде. Следовательно, правила оформления кода должны быть очень строго определены. А рекомендации по программированию могут зачастую иметь намного более серьезное воздействие на безопасность и надежность, просто потому, что их можно определить менее строго, чем правила.

Во многих из предлагаемых здесь рекомендаций делаются ссылки на правила, излагаемые в книге *The CERT® Oracle® Secure Coding Standard for Java™*, описывающей одноименный стандарт, в такой форме: “IDS01-J. Нормализуйте символьные строки перед проверкой их достоверности”, где первые три буквы обозначают соответствующую главу данной книги. В частности, IDS обозначает главу 2 “Input Validation and Data Sanitization” (IDS — Проверка достоверности и санобработка данных).

Правила из стандарта безопасного программирования на Java доступны также на веб-сайте, посвященном вопросам безопасного программирования (`www.securecoding.cert.org`), где они продолжают совершенствоваться. В стандарте, изложенном в книге *The CERT® Oracle® Secure Coding Standard for Java™*, дается определение правил для целей тестирования на соответствие, а на упомянутом выше сайте можно найти дополнительные сведения или ценные выводы, отсутствующие в данной книге, но помогающие лучше интерпретировать смысл этих правил. Перекрестные ссылки на другие рекомендации делаются в тексте книги с указанием номера и названия соответствующей рекомендации.

Предмет книги

Эта книга посвящена вопросам программирования на платформе Java SE 7 и содержит рекомендации для написания безопасного кода с помощью прикладного программного интерфейса API для версии Java SE 7. В спецификации на язык программирования Java в версии Java SE 7 [JLS 2013] регламентируется поведение этого языка, и поэтому она послужила основным источником для разработки рекомендаций, представленных в данной книге.

В стандарты на такие традиционные языки программирования, как C и C++, включается описание неопределенных, непредусмотренных и определенных в реализации видов поведения, которые могут привести к появлению уязвимостей, когда программист делает неправильные предположения относительно переносимости этих видов поведения. Напротив, в спецификации на Java подобные виды поведения определены более полно, поскольку язык программирования Java разработан как независимый от конкретной платформы. Но и в этом случае некоторые виды поведения отдаются на откуп реализаторам виртуальной машины Java (JVM) или компилятора Java. Именно такие языковые особенности учитываются в представленных здесь рекомендациях, предлагающих реализаторам решение насущных вопросов и помогающих программистам правильно оценить и уяснить присущие языку ограничения, чтобы найти способы их преодоления.

Для того чтобы писать надежные и безопасные программы, недостаточно уделить внимание вопросам одного только языка программирования. Иногда вопросы проектирования, возникающие при обращении к прикладным программным интерфейсам API языка Java, приводят к тому, что эти интерфейсы становятся не рекомендуемые к дальнейшему применению. А порой интерфейсы API или документация на них интерпретируются программистами неверно. На подобные неясности в интерфейсах API указывается в представленных здесь рекомендациях, где обращается внимание на правильное их применение. Эти рекомендации дополняются характерными образцами ошибочного проектирования или неверно выбранного стиля программирования.

Ядро языка Java и его расширения в виде прикладных интерфейсов API, а также виртуальная машина JVM предоставляют ряд средств для обеспечения безопасности, в том числе диспетчер защиты и контроллер доступа, шифрование, автоматическое управление памятью, строгий контроль типов и проверку достоверности байт-кода. Эти средства обеспечивают достаточную безопасность для большинства приложений, но при условии, что они правильно используются, что очень важно. В представленных здесь рекомендациях указываются скрытые препятствия и предупреждаются опасности, связанные с архитектурой системы безопасности, а также делается акцент на правильную ее реализацию. Придерживаясь этих рекомендаций, можно уберечься от многих программных ошибок, используемых для нарушения защиты с целью отказать в обслуживании, организовать утечку информации или намеренно превысить полномочия.

Рассматриваемые библиотеки

На рис. В.1 приведена концептуальная схема программных продуктов компании Oracle на платформе Java SE. Представленные здесь рекомендации по программированию направлены на разрешение вопросов, связанных, главным образом, с базовыми библиотеками, в том числе `lang` и `util`. Они позволяют избежать характерных программных ошибок, которые уже исправлены или не имеют отрицательных последствий. А о функциональных программных ошибках упоминается лишь в тех случаях, если они происходят часто, представляют серьезную угрозу для безопасности и надежности или оказывают отрицательное воздействие на большинство технологических средств Java, опирающихся на базовую платформу. Эти рекомендации не ограничиваются только вопросами безопасности, характерными для базового прикладного интерфейса API, но и касаются важных вопросов надежного и безопасного применения стандартных расширений этого прикладного интерфейса API (в пакете `javax`).

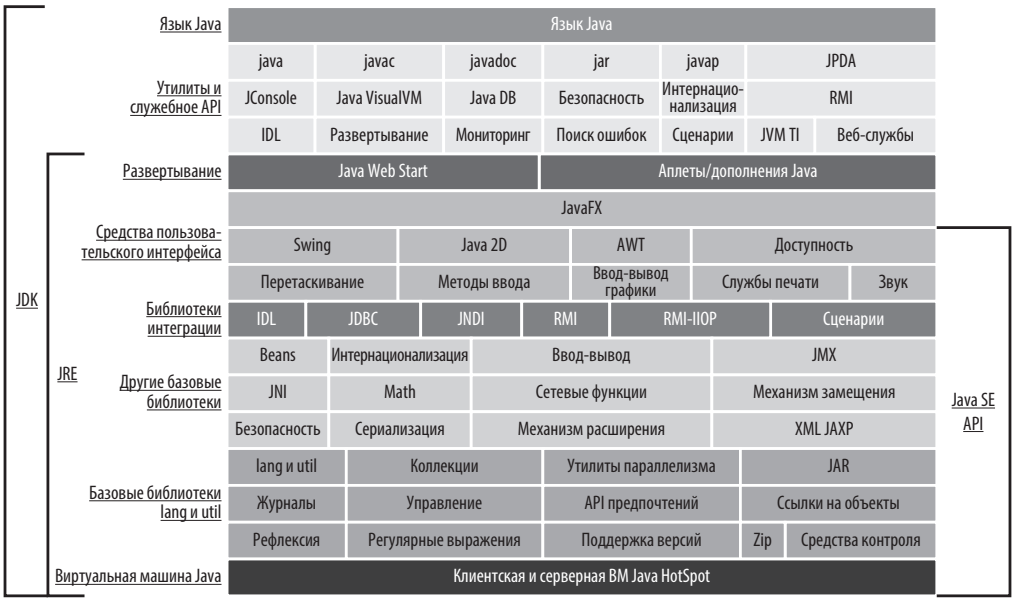

Рис. В.1. Концептуальная схема программных продуктов компании Oracle на платформе Java SE. (Из документации на платформу Java SE компании Oracle, `http://docs.oracle.com/javase/7/docs/`. Copyright © 1995, 2010, компания Oracle и все ее филиалы. Все права защищены)

Для демонстрации всего спектра средств, предлагаемых в Java для обеспечения безопасности, требуется исследовать взаимодействие кода с другими компонентами и каркасами приложений. В представленных здесь рекомендациях по программированию на Java иногда приводятся примеры, взятые из распространенных каркасов настольных и веб-приложений, включая Spring и Struts, а также технологии вроде Java Server Pages (JSP), чтобы показать уязвимость, которую нельзя рассматривать в отдельности. А библиотеки и прочие сторонние решения предлагаются лишь в тех случаях, когда в стандартном прикладном интерфейсе API отсутствуют средства для полного или частичного устранения уязвимости.

Вопросы, не рассматриваемые в этой книге

В представленных здесь рекомендациях по безопасному программированию на Java не рассматриваются следующие вопросы.

- **Содержимое.** Представленные здесь рекомендации по программированию распространяются на все платформы, но не охватывают вопросы, касающиеся только одной платформы на Java. В частности, рекомендации для платформы Android, Java Micro Edition (ME) или Java Enterprise Edition (EE), как правило, не годятся для Java Standard Edition (SE). Этими рекомендациями не охватываются также прикладные интерфейсы API в версии Java SE, поддерживающие воспроизведение звука и графики, управление доступом по учетным записям пользователей, организацию сеансов работы, аутентификацию в пользовательском интерфейсе (и наборе инструментальных средств для него) или веб-интерфейсе. Тем не менее в этих рекомендациях обсуждаются сетевые системы на Java в отношении тех рисков, которые связаны с неправильной проверкой достоверности вводимых данных и внесением дефектов, а также предлагаются подходящие методики для устранения подобных недостатков. В этих рекомендациях предполагается, что функциональные требования к программному продукту правильно выявляют и предотвращают уязвимости на более высоком уровне проектирования и разработки архитектуры.
- **Стиль программирования**. Вопросы стиля программирования носят довольно субъективный характер, и поэтому выработать и дать подходящие рекомендации по этим вопросам не представляется возможным. Следовательно, в этой книге, как правило, соблюдение любого отдельно взятого стиля программирования не затрагивается. Вместо этого читателям рекомендуется выработать свои руководящие принципы относительно стиля программирования и придерживаться их неукоснительно. Проще всего придерживаться выбранного стиля программирования с помощью инструментального средства для форматирования исходного кода. Такие средства предоставляются во многих интегрированных средах разработки (ИСР).
- **Инструментальные средства.** Многие из упоминаемых здесь рекомендаций непригодны для автоматического выявления или исправления ошибок. В некоторых случаях поставщики инструментальных средств могут реализовать проверку для выявления нарушений этих рекомендаций. Но Финансируемый государством научно-исследовательский центр (FFRDC) и Институт программотехники (SEI) не рекомендуют для этой цели конкретные инструментальные средства или их поставщиков.
- **Противоречивые рекомендации.** В целом из представленных здесь рекомендаций исключены противоречивые и не нашедшие общего признания рекомендации.

Кому адресована книга

Эта книга адресована главным образом разработчикам программного обеспечения на Java. И хотя представленные в ней рекомендации касаются платформы Java SE 7, они могут пригодиться (хотя и не полностью) разработчикам, работающим на платформе Java ME или Java EE, а также тем, кто пользуется другими версиями Java.

Несмотря на то что эти рекомендации предназначены для построения надежных и безопасных систем, они окажутся полезными и для достижения других качеств, в том числе защищенности, безотказности, устойчивости, работоспособности и сопровождаемости. Эти рекомендации могут быть использованы следующими категориями специалистов.

- Разработчики инструментальных средств анализа, стремящиеся диагностировать программы, небезопасные или несоответствующие нормам языка Java.
- Руководители групп разработчиков программного обеспечения, заказчики программного обеспечения и прочие специалисты как со стороны исполнителей, так и со стороны заказчиков программного обеспечения, которым требуется установить строгие нормы на безопасное программирование.
- Преподаватели, составляющие курсы программирования на Java.

Содержание и организация книги

Эта книга состоит из 75 рекомендаций, организованных по главам следующим образом.

- **Глава 1, "Безопасность".** В этой главе представлены рекомендации по обеспечению безопасности прикладных программ на Java.
- **Глава 2, "Защитное программирование".** Эта глава содержит рекомендации по защитному программированию, позволяющие писать код, который сам себя защищает от всяких неожиданностей.
- **Глава 3 "Надежность".** В этой главе даются рекомендации по повышению надежности и безопасности прикладных программ на Java.
- **Глава 4, "Понятность программ".** В этой главе даются рекомендации, позволяющие сделать программы более понятными и удобочитаемыми.
- **Глава 5, "Ложные представления программистов".** В этой главе рассматриваются характерные случаи, когда возникают недоразумения и ложные представления при программировании на Java.

В приложении А описывается применимость представленных в книге рекомендаций к разработке прикладных программ на Java для платформы Android. А в приложениях Б и В приводится словарь общеупотребительных терминов и библиографический перечень соответственно.

Представленные в книге рекомендации имеют согласованную структуру. В названии и вводном абзаце определяется сущность рекомендации. Далее обычно следует один или более пример кода, не соответствующий принятым нормам безопасного, надежного, понятного и корректного программирования на Java, а также решения, позволяющие привести код к этим нормам. Каждая рекомендация завершается разделом, посвященным ее применимости и библиографическими ссылками на нее.