Требования к надежности и функциональной безопасности программных продуктов

Программа любой сложности и назначения при строго фиксированных исходных данных и абсолютно надежной аппаратуре исполняется по однозначно определенному маршруту и дает на выходе строго определенный результат. Однако случайное изменение исходных данных и накопленной при обработке информации, а также множество условных переходов в программе создают огромное число различных маршрутов исполнения каждого сложного комплекса программ. Источниками ненадежности являются непроверенные тестированием сочетания исходных данных, при которых функционирующие программы дают неверные результаты или отказы. В результате комплекс программ не соответствует требованиям функциональной пригодности и работоспособности.

В международном стандарте ISO/IEC 9126 при отборе минимума стандартизируемых показателей выдвигаются и учитываются следующие принципы: ясность и измеряемость значений, отсутствие перекрытия между используемыми показателями, соответствие установившимся понятиям и терминологии, возможность последующего уточнения и детализации.

Надежность — способность комплекса программ обеспечивать достаточно низкую вероятность потери работоспособности (отказа) в процессе функционирования программного продукта в реальном времени.

Основные атрибуты надежности могут быть объективно измерены и сопоставлены с требованиями.

Завершенность — способность комплекса программ не попадать в состояния отказов вследствие ошибок и дефектов в программах, данных и внешней среде.

Устойчивость к дефектам и ошибкам — способность программного продукта автоматически поддерживать заданный уровень качества функционирования при проявлениях дефектов и ошибок или нарушениях установленного интерфейса между компонентами и с внешней средой.

Восстанавливаемость — способность комплекса программ в случае отказа возобновлять требуемый уровень качества функционирования, поврежденные программы и данные.

После отказа программный продукт иногда может быть неработоспособным в течение некоторого времени, продолжительность которого определяется его восстанавливаемостью в процессе перезапуска. Перезапуск должен обеспечивать возобновление нормального функционирования ПС; для этого требуются ресурсы ЭВМ и время. Поэтому полнота и длительность восстановления функционирования после сбоев отражают качество и надежность программного продукта и возможность его использования по прямому назначению.

Доступность или готовность — способность программного продукта выполнять требуемую функцию в данный момент времени при заданных условиях использования.

Готовность может оцениваться относительным временем, в течение которого комплекс программ находится в работоспособном состоянии, в пропорции к общему времени применения.

Понятие корректной (правильной) программы может рассматриваться статически, вне ее исполнения во времени. Корректность программы не определена вне области изменения исходных данных, заданных требованиями спецификации, и не зависит от динамики функционирования программы в реальном времени. Степень некорректности программ определяется вероятностью попадания реальных исходных данных в пространство значений, которое задано требованиями спецификации и технического задания, однако не было проверено при тестировании и испытаниях.

Надежная программа прежде всего должна обеспечивать достаточно низкую вероятность отказа в процессе функционирования в реальном времени. Быстрое реагирование на искажения программ, данных или вычислительного процесса и восстановление работоспособности за время, меньшее, чем порог между сбоем и отказом, обеспечивают высокую надежность программ.

При оценке реализации требований надежности регистрируются только такие искажения в процессе динамического исполнения программ, которые приводят к потере работоспособности продукта. Реальные исходные данные могут иметь значения, отличающиеся от заданных требованиями и от использованных при применении программ. Непредсказуемость вида, места и времени проявления дефектов в процессе эксплуатации приводит к необходимости создания специальных, дополнительных, систем оперативной защиты от непредумышленных, случайных искажений вычислительного процесса, программ и данных.

Основным принципом классификации сбоев и отказов в программных продуктах при отсутствии их физического разрушения является разделение по временно́му показателю длительности восстановления после любого искажения программ, данных или вычислительного процесса, регистрируемого как нарушение работоспособности.

При длительности восстановления, меньшей заданного порога, дефекты и аномалии при функционировании программ следует относить к сбоям, а при восстановлении, превышающем по длительности пороговое значение, происходящее искажение соответствует отказу. Временная зона перерыва нормальной выдачи информации и потери работоспособности, которую следует рассматривать как зону сбоя, тем шире, чем более инертный объект находится под воздействием сообщений, подготовленных программным продуктом.

Требования к надежности программных продуктов в значительной степени адекватны аналогичным характеристикам, принятым для других технических систем. Наиболее широко используется критерий длительности наработки на отказ. Для определения этой величины измеряется время работоспособного состояния системы между последовательными отказами или началами нормального функционирования системы после них. Критерий надежности восстанавливаемых систем учитывает возможность многократных отказов и восстановлений. Для оценки надежности таких систем, которыми чаще всего являются сложные программные продукты, кроме вероятностных характеристик наработки на отказ важную роль играют характеристики функционирования после отказа в процессе восстановления.

Основными требованиями к процессу восстановления являются длительность восстановления и ее вероятностные характеристики. Этот критерий учитывает возможность многократных отказов и восстановлений. Обобщение характеристик отказов и восстановлений производится в критерии «коэффициент готовности». Этот показатель отражает вероятность наличия восстанавливаемой системы в работоспособном состоянии в произвольный момент времени. Значение коэффициента готовности соответствует доле времени полезной работы системы на достаточно большом интервале, содержащем отказы и восстановления.

В реальных системах требуемая наработка на отказ программного продукта обычно должна быть соизмерима с наработкой на отказ аппаратуры, на которой исполняется программа. Для систем обработки информации и управления в реальном времени наработка на отказ комплекса программ измеряется десятками и сотнями часов, а для особо важных или широко тиражируемых систем она может достигать десятков тысяч часов.

Наиболее полно функциональная безопасность комплексов программ характеризуется величиной ущерба, возможного при проявлении дестабилизирующих факторов и реализации конкретных угроз — рисков, а также средним временем между проявлениями непредумышленных угроз, нарушающих безопасность. Однако описать и измерить в общем виде возможный ущерб при нарушении безопасности для критических ПС разных классов практически невозможно. Поэтому реализации угроз можно характеризовать интервалами времени между их проявлениями, нарушающими безопасность применения ПС, или наработкой на отказы, отражающиеся на безопасности. Это сближает понятия и требования безопасности с показателями надежности комплексов программ.

Различие заключается в том, что в показателях надежности учитываются все реализации отказов, а в характеристиках функциональной безопасности следует регистрировать только те случайные катастрофические отказы, которые отразились на безопасности. Статистически таких отказов может быть в несколько раз меньше, чем учитываемых в значениях надежности. Однако методы, влияющие факторы и реальные значения показателей надежности могут служить ориентирами при оценке функциональной безопасности критических программных продуктов.

При формировании требований к средствам обеспечения функциональной безопасности программных продуктов систем необходимо учитывать, что такие средства не могут быть абсолютно безупречными и корректными. Непредусмотренные при проектировании некоторые ситуации и ошибки функционирования программ и данных могут быть потенциальными источниками катастроф при применении таких программных продуктов, влияющими на безопасность их функционирования и применения.

Поэтому при подготовке требований к безопасности программных продуктов целесообразно учитывать и конкретизировать особенности источников возможного нарушения корректности их функционирования:

- дефекты методов, алгоритмов и функционального содержания процессов, принятых для решения задач обеспечения безопасности для корректного применения системы по ее назначению;
- недостаточное качество технологии и производственных процессов, использованных для реализации методов и алгоритмов обеспечения безопасности, определяющих возможность проявления технологических дефектов и ошибок при применении программного продукта;

- нарушения внешней средой, системами и (или) пользователями требований к созданным средствам обеспечения безопасного функционирования и применения программных продуктов.

В результате при использовании программного продукта по прямому назначению в реальной внешней среде, при проявлениях ошибок, искажениях исходных данных и других непредсказуемых событиях возможны опасные ситуации отказа, аномалии функционирования и искаженные результаты, нарушающие безопасность системы. Предвидеть заранее все подобные ситуации и протестировать при них сложные ПС оказывается невозможным из-за их огромного количества. Один из видов ущерба при возникновении отказов, определяющих функциональную безопасность, заключается в прерывании его работоспособности на длительное время. Контроль продолжительности потенциальной работоспособности системы и ПС, а также длительности отказа может требовать повышения функциональной безопасности путем автоматизированного сокращения времени неработоспособного состояния системы.

Отказы становятся катастрофическими и отражаются на безопасности, если длительность прерывания работоспособного состояния превышает некоторое пороговое значение, специфическое для конкретной системы и внешней среды. Если удается обнаружить опасную ситуацию отказа и восстановить работоспособность за время, не превышающее заданный порог, то эта ситуация может не отразиться на функциональной безопасности системы. Для этого применяются методы и средства, направленные на автоматическое обнаружение опасных отказов при реальном функционировании комплексов программ и на достаточно быстрое оперативное автоматическое восстановление нормального вычислительного процесса, текстов программ и данных (рестарт). В любых ситуациях прежде всего должны исключаться катастрофические последствия и длительные опасные отказы или в максимальной степени смягчаться их влияние на безопасность результатов.

Введение в программы средств контроля и оперативной защиты позволяет скомпенсировать их неполную корректность, а также снижать влияние на безопасность негативных возмущений различных типов. Выбор метода оперативного восстановления обычно происходит в условиях значительной неопределенности сведений о характере отказа и степени его возможного влияния на работоспособность ПС. Кроме того, восстановление работоспособности желательно производить как можно быстрее, чтобы отказ можно было свести до уровня кратковременного сбоя.

Требования к обеспечению функциональной безопасности программных продуктов установлены в международном стандарте ISO/IEC 61508-3:2012. В нем представлен общий подход к видам деятельности на протяжении цикла обеспечения безопасности для систем, содержащих программируемые электронные компоненты систем (ПЭС), которые используются для выполнения различных функций, в частности для обеспечения безопасности систем.

Стандарт ISO/IEC 61508-3:2012 содержит:

- метод разработки спецификаций требований по безопасности ПЭС, необходимых для достижения заданной функциональной безопасности;
- основанный на рисках подход для определения требований к уровням соответствия комплексу требований на функциональную безопасность;
- количественные меры отказов для систем безопасности ПЭС, связанные с уровнями соответствия комплексу требований систем по функциональной безопасности;
- требования для этапов жизненного цикла обеспечения безопасности и деятельности, которая должна проводиться при проектировании и разработке программного продукта, обеспечивающего безопасность;
- требования для информации, относящейся к аттестации программных продуктов, которая должна передаваться организациям, производящим компоновку ПЭС в составе системы;
- требования, которые должны выполнять организации, производящие модификацию программного продукта, обеспечивающего безопасность;
- требования к технологическим средствам производства, таким как инструментальные средства проектирования и разработки, языковые трансляторы, инструментальные средства для тестирования, поиска и устранения ошибок, средства управления конфигурацией.

Цикл обеспечения требований безопасности при разработке комплекса программ должен быть выбран и задан при планировании безопасности для практических нужд проекта или предприятия. В деятельность, связанную с циклом обеспечения безопасности, должны быть включены процедуры гарантирования качества и безопасности. Рекомендуется использовать основные положения, комплекс этапов и работ, представленный в стандарте ISO/IEC 12207:2010.

Спецификация требований по безопасности программного продукта должна быть достаточно подробной, для того чтобы конструкция и ее выполнение достигали

требуемого соответствия комплексу требований по безопасности, и можно было произвести оценку функциональной безопасности.

В спецификации должны быть представлены требования к функциям безопасности:

- обеспечивающие достижение и поддержание безопасного состояния программного продукта или системы;
- относящиеся к выявлению отказов; извещения об их наличии и управление отказами в аппаратуре, отказами исполнительных механизмов, а также в программном продукте;
- относящиеся к периодической проверке функций обеспечения безопасности в оперативном режиме и отключенном состоянии.

Аттестация программного продукта должна быть использована для демонстрации того, что программный продукт удовлетворяет требованиям по его безопасности. План аттестации безопасности программного продукта должен содержать требуемые условия окружающей среды, в которой должна производиться аттестация; политику и процедуры оценки результатов аттестации.

Требования к информационной безопасности комплексов программ формализуют стандарты ISO/IEC 15408-1:2012, 15408-2:2013, 15408-3:2013. Они состоят из трех частей, отражающих требования и рекомендации по обеспечению безопасности систем, содержащих программные средства. Первая часть определяет концепцию всего стандарта. Вторая, самая большая, часть формализует методы обеспечения безопасности. Третья часть полностью посвящена требованиям и процессам обеспечения доверия (качества) компонентов систем, реализующих функции их безопасности. Положения этой части стандарта трактуются с позиции обеспечения функциональной безопасности, а термин «доверие» означает качество или уверенность выполнения методических и технологических требований безопасности.

Оценка и утверждение целей функциональной безопасности требуются для демонстрации заказчику или пользователю, что установленные цели проекта адекватны методам обеспечения его безопасности. Существуют цели и функции безопасности для ПС и цели безопасности для среды. Рекомендуется сопоставлять эти цели безопасности с идентифицированными угрозами, которым они противостоят, и (или) с политикой и предположениями, которым они должны соответствовать. Использование стандарта означает, что требования могут быть четко идентифицированы, что

они автономны и применение каждого требования возможно и даст значимый результат при оценке качества, основанный на анализе соответствия продукта этому конкретному требованию.

Для систем реального времени особое значение имеют требования эффективного использования программным продуктом ресурсов ЭВМ по производительности. Эффективность в стандарте ISO/IEC 9126:93 отражена двумя динамическими характеристиками требований: временной эффективностью и используемостью ресурсов ЭВМ, которые рекомендуется описывать в основном количественными атрибутами, учитывающими динамику функционирования комплексов программ. В этих стандартизированных характеристиках отражается только частная, конструктивная эффективность использования ресурсов ЭВМ, которую не следует смешивать с требованием системной эффективности функциональной пригодности программного продукта при применении в конкретной системе.

Основные требования к характеристикам эффективности использования вычислительных ресурсов сосредоточены на наиболее критичных показателях производительности и длительности решения функциональных задач. При этом в контракте, техническом задании и спецификации требований должны быть зафиксированы и утверждены требуемые значения этих характеристик и их приоритетов. В стандарте ISO/IEC 9126:93 эти характеристики качества комплексов программ рекомендуется отражать рядом атрибутов, каждый из которых следует оценивать для средних и наихудших сценариев функционирования комплекса программ.

Временная эффективность — свойства комплекса программ, отражающие требуемое время обработки заданий; время отклика из ЭВМ в систему и (или) внешнюю среду после получения типового задания и начала решения требуемой функциональной задачи; производительность решения задач с учетом количества используемых вычислительных ресурсов (ISO/IEC 14756).

Временная эффективность программного продукта определяется длительностью выполнения заданных функций и ожидания результатов в средних и (или) наихудших случаях с учетом приоритетов задач. Пропускная способность решения функциональных задач — производительность, т. е. число заданий, которое можно реализовать на данной ЭВМ в заданном интервале времени в зависимости от их содержания и числа действующих пользователей или воздействий из внешней среды.

Используемость ресурсов — степень загрузки доступных вычислительных ресурсов в течение заданного времени при выполнении функций комплекса программ в установленных условиях.

Ресурсная экономичность отражается занятостью ресурсов центрального процессора, оперативной, внешней и виртуальной памяти, каналов ввода—вывода, терминалов и каналов сетей связи исполнением программ.