

ФОРМАЛИЗАЦИЯ ФУНКЦИЙ БЕЗОПАСНОСТИ И ОБЕСПЕЧЕНИЕ КАЧЕСТВА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ СИСТЕМ, ВАЖНЫХ ДЛЯ БЕЗОПАСНОСТИ АЭС

Жарко Е.Ф.

Институт проблем управления им. В.А. Трапезникова РАН,
Россия, г. Москва, ул. Профсоюзная д.65
zharko@ipu.ru

Аннотация: Одной из важнейших задач в обеспечении качества программно-технических комплексов является процедура формирования требований к разрабатываемой или модифицируемой системе и последующая их верификация. Наиболее существенные ошибки совершаются на первых фазах жизненного цикла – это ошибки в определении требований, выборе архитектуры, высокоуровневом проектировании. Отказы критического по отношению к безопасности программного обеспечения могут нанести серьезный ущерб оборудованию или свойствам, а также привести к существенному ущербу окружающей среде или к человеческим жертвам. Увеличение требований к качеству программного обеспечения для систем важных для безопасности АЭС на всех этапах жизненного цикла связано с возрастанием сложности и функциональности программного обеспечения и привело к необходимости разработки подходов для обоснования как безопасности самих систем, так и входящего в их состав программного обеспечения. В статье рассматривается подход, основанный на построении “функций безопасности” выполнение которых в дальнейшем верифицируется.

Ключевые слова: программное обеспечение, функция безопасности, обеспечение качества, верификация, валидация, АЭС.

Введение

Развитие автоматизации объектов критической инфраструктуры с повышенным риском эксплуатации, в том числе и в атомной энергетике, характеризуется тенденцией разработки автоматизированных систем управления технологическими процессами (АСУ ТП), реализующих значительно более сложные алгоритмы управления и анализа данных с использованием сложных программно-технических комплексов (ПТК) [1-4]. Разработка ПТК, их верификация и валидация, а также последующая их эксплуатация, а стечением времени, и модернизация, должны соответствовать и удовлетворять принятому уровню безопасности.

С возрастанием требований к объектам критической инфраструктуры, сложность программного обеспечения и его важность в обеспечении функций всей системы резко возрастает. Программное обеспечение (ПО) играет все более важную роль в выявлении и контроле опасных факторов, а также в критических по отношению к безопасности функциях [5, 6]. Широкое распространение программно-технических систем для объектов с повышенным риском эксплуатации привело к необходимости разработки методов для обоснования безопасности таких систем. При обосновании безопасности в существующих подходах [7-9] использование моделей качества и безопасности играет центральную роль. Обеспечение качества программного обеспечения АСУ ТП на всех этапах его жизненного цикла базируется на качественном и количественном анализе, который, согласно нормативной документации, также должен проводиться на всех этапах. Качественный и

количественный анализ качества программного обеспечения должен учитывать две составляющие программно-технических комплексов: аппаратную и программную [10].

1 Качество программного обеспечения и особенности определения функции безопасности

Программное обеспечение вносит существенный вклад в функции, выполняемые системами важными для безопасности. Программное обеспечение может поддерживать дополнительные функции, введенные в соответствии с проектом разрабатываемой или уже функционирующей системы. Для систем важных для безопасности АЭС жизненный цикл безопасности программного обеспечения тесно связан с жизненным циклом безопасности самой системы. Спецификация требований к программному обеспечению является частью спецификации системы.

Разработанный метод комплексной верификации программного обеспечения [11] основан на учете требований стандартов по безопасности, интегрирует этапы верификации ПО и их атрибуты, включая задействованный персонал, задачи, методики, устранение недостатков и выпускаемую документацию. Эффективность метода комплексной верификации программного обеспечения была подтверждена в ходе работ по разработке информационных и управляющих систем, важных для безопасности АЭС. Определение и выбор функций безопасности имеет ряд особенностей и проблем. Во время анализа ПО на безопасность может оказаться, что принятая функция безопасности не является необходимой или достаточной, когда заданная функция безопасности слишком строга и доказательство безопасности невозможно, или напротив, слишком слаба, из-за чего уменьшается вероятность нахождения ошибок ПО. Одним из возможных способов повышения качества ПО в рамках применяемого комплексного подхода является доказательство корректности, которое относится к формальным методам [12, 13].

2 Общие требования к формализации функции безопасности

С точки зрения доказательства корректности все рассматриваемые понятия (свойства, функции и пр.) должны быть формализованы, так как в противном случае доказать что-либо с помощью любых формальных методов будет невозможно. Опыт верификации систем важных для безопасности АЭС выявил ряд ситуаций, когда задание функций безопасности в формализованном виде для некоторых свойств не представляется возможным.

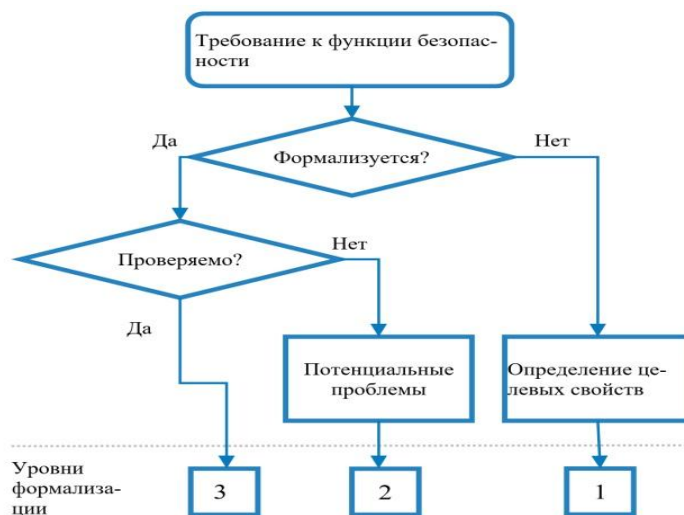


Рис. 1. Определение уровня формализации требования к безопасности

В этом случае решением является формализованное описание свойств рассматриваемого понятия в качестве задачи доказательства корректности и определения функции безопасности, а в дальнейшем делается экспертное заключение о том, является ли верифицированное свойство безопасным или нет. Для этого сначала посредством доказательства корректности определяются формализованные свойства системы, а в последующем, на их основании, делается вывод о безопасности системы. Используем три уровня формализации: 1. неформализованный, 2. формализованный, 3. проверяемый. Определение уровня формализации можно в соответствии с алгоритмом, представленным на рис. 1.

Первый уровень является вербальной формулировкой. Его недостатком является то, что необходимый впоследствии переход к формальному уровню неоднозначен, что может вести к

проблемам безопасности и сложностям при доказательстве. Поэтому переход ко второму уровню делать необходимо, и выполнять его как можно раньше.

Неформализованный уровень появляется изначально при формулировке, удобен в общении, не требует существенных затрат и является абстрактным. Кроме того, он широко используется в нормативных документах. Однако в случае рассмотрения конкретной системы и доказательства ее корректности требуется формализация, которая устраняет неоднозначности, улучшает понимание и может быть помещена в некоторую формальную систему для последующего доказательства корректности. Доказательство корректности может работать со вторым уровнем формализации, но отсутствие возможности проверки или её ограниченность сигнализирует о потенциальных проблемах, так как возможно будет затруднительно использовать другие способы верификации, такие как тестирование, имитационные испытания и др. Сама же необходимость перехода на третий уровень согласуется с опытом создания надежных и безопасных систем.

Заключение

Для ПТК, используемых в системах важных для безопасности АЭС, существует задача обеспечения правильного (в отношении спецификации), безопасного и полного выполнения требований. Обоснование безопасности системы, безопасности и целостности определенного программного обеспечения основывается на проектировании и проектных документах, представленных во время разработки системы, результате анализа спецификаций, алгоритмов и реализации. Подход, к определению функций безопасности, представленный в статье, был применен и позволил повысить качество разрабатываемого/ модифицированного программного обеспечения:

- при верификации программного обеспечения систем верхнего уровня АСУ ТП АЭС, относящихся к системам важным для безопасности;
- для выявления ошибок проектирования программного обеспечения на ранних стадиях разработки с целью снижения рисков возникновения нештатных ситуаций в процессе эксплуатации объектов;
- при обосновании качества программного обеспечения систем важных для безопасности АЭС на всех этапах жизненного цикла,

Литература

1. *Poletykin A., Jharko E., Mengazetdinov N., Promyslov V.* Some Issues of Creating the New Generation of Upper Level Control Systems of NPP APCS // Proceedings of the 5th International Conference on Control, Instrumentation, and Automation (ICCIA 2017, Shiraz, Iran). 2017. – P. 78-83.
2. *Менгазетдинов Н.Э., Полетыкин А.Г., Бывайков М.Е., Промыслов В.Г., Жарко Е.Ф., Смирнов В.Б., Акафьев К.В.* Автоматизация атомных электростанций – опыт ИПУ РАН // Труды XII Всероссийского совещания по проблемам управления ВСПУ-2014. Москва, 16-19 июня 2014 г. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2014. – С. 4219-4236.
3. *Коган И.Р., Полетыкин А.Г., Промыслов В.Г., Жарко Е.Ф.* Эволюция АСУТП АЭС для ВВЭР, проблемы, нерешенные вопросы, новые угрозы и возможные направления развития // Труды XII Всероссийского совещания по проблемам управления ВСПУ-2014. Москва, 16-19 июня 2014 г. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2014. – С. 4200-4211.
4. *Бывайков М.Е., Жарко Е.Ф., Менгазетдинов Н.Э., Полетыкин А.Г., Прангишвили И.В., Промыслов В.Г.* Опыт проектирования и внедрения системы верхнего блочного уровня АСУ ТП АЭС // Автоматика и телемеханика. 2006. №. 5. – С. 65-79.
5. *Restu Maeran, Joyce Kemunto Mayaka, Jae Cheon Jung.* Software verification process and methodology for the development of FPGA-based engineered safety features system Author links open overlay panel // Nuclear Engineering and Design. Vol. 330. 2018. – P. 325-331.
6. *Ye Cheng, Ni Chao, Zheng Tian, Zhang Zhicheng, Zhang Ronghua.* Quality assurance for a nuclear power plant simulator by applying standards for safety-critical software // Progress in Nuclear Energy. Vol. 70. 2014. – P. 128-133.
7. *Leveson N.G., Cha S.S., Shimeall T.J.* Safety verification of ADA programs using software fault trees // IEEE Software. 1991. Vol. 8, No. 4. P. 48-59.
8. *Bozzano M., Villaflorita A., Kerlund O., Bieber P., Bounol C., Bde E., et. al.* ESACS: An integrated methodology for design and safety analysis of complex systems // Proceedings of the European Safety and Reliability Conference (ESREI 2003). – P. 237-245.
9. *Жарко Е.Ф.* Проблемы управления качеством программного обеспечения // Труды II Международной конференции “Идентификация систем и задачи управления” SICPRO '03. Москва, 29 -31 января 2003 г. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2003. – С. 887-923.
10. *Smith D., DeLong T., Johnson B.W.* A Safety Assessment Methodology for Complex Safety-Critical Hardware/Software Systems // International Topical Meeting on Nuclear Plant Instrumentation, Controls, and Human-Machine Interface Technologies. Washington, DC, November, 2000.

11. *Jharko E.Ph.* Towards Quality Assurance under Developing Safety Important Systems Software for Nuclear Power Plants // Proceedings of 2018 International Russian Automation Conference (RusAutoCon). IEEE, 2018. – P.1-6.
12. *Souri A., Navimipour N.J., Rahmani A.M.* Formal verification approaches and standards in the cloud computing: A comprehensive and systematic review // Computer Standards & Interfaces. Vol. 58. 2018. – P. 1-22.
13. *Linna Pang, Chen-Wei Wang, Mark Lawford, Alan Wassynq.* Formal verification of function blocks applied to IEC 61131-3 // Science of Computer Programming. Vol. 113, Part 2. 2015. – P. 149-190.