УДК 621.34

Т.Н. Шипова, В.В. Босько, И.А. Березюк

Национальный технический университет «ХПИ», Харьков

АНАЛИЗ ТРЕБОВАНИЙ ИНФОРМАЦИОННОЙ И ФУНКЦИОНАЛЬНОЙ БЕЗОПАСНОСТИ К ОПЕРАЦИОННЫМ СИСТЕМАМ РЕАЛЬНОГО ВРЕМЕНИ

В работе проведен анализ отличительных особенностей операционных систем реального времени, требований информационной и функциональной безопасности к этим системам в условиях использования их в проблемно-ориентированных компьютерных комплексах. Представлен обзор основных отраслей, в которых могут быть использованы операционные системы реального времени. Сделаны выводы о необходимости выполнения заданных требований к операционным системам реального времени и использования методов и средств обеспечения информационной и функциональной безопасности.

Ключевые слова: *операционные системы реального времени, безопасность, требования информационной и функциональной безопасности.*

Введение

Постановка задачи. В настоящее время операционные системы реального времени (ОСРВ) используются во многих ключевых отраслях жизнеобеспечения современного общества. Они отвечают за работу систем транспортной инфраструктуры, авиапромышленности, легкой и тяжелой промышленности, атомных и космических станций. В связи с этим в работе ОСРВ должны отвечать жестким требованиям пропускной способности, реакции на внешние воздействия, надежности и безопасности. Вопросу безопасности в компьютерной сфере, в том числе и при работе в ОСРВ в настоящее время во всем мире уделяется достаточно большое внимание. Проблемно-ориентированная направленность ОСРВ предполагает определенную ответственность за безопасную работу системы, не предполагая появления критических ситуаций. Субъект, осуществляющий атаку на проблемно-ориентированную ОСРВ, может вызвать сбой системы и вызвать своими действиями катастрофу локального или даже глобального масштаба (например, критический «скачок» напряжения в агрегатах атомной промышленности). Анализ литературы показал, что вопросам предупреждения угроз информационной и функциональной безопасности в ОСРВ, исходя из соответствующих требований, уделяется недостаточно внимания. Поэтому задача анализа требований информационной и функциональной безопасности к ОСРВ остается актуальной.

Результаты анализа

Проведенный анализ литературы [1-7] показал, что ОСРВ имеют ряд ключевых отличий по сравнению с общераспространенными операционными системами (ОС). В табл. 1 представлены основные отличия между ОСРВ и ОС. ОСРВ – система, которая должна реагировать на множество событий, пользуясь ограниченными ресурсами. Главным ресурсом, отвечающим за распределение между несколькими задачами (называется диспетчеризацией – scheduling), является процессор. Поэтому в однопроцессорной системе по-настоящему параллельное выполнение нескольких задач невозможно [2]. Для этого существует достаточно большое количество различных методов диспетчеризации.

Таблица 1

Основные отличия между ОСРВ и ОС

	ОСРВ	ОС общего назначения
Основная задача	Своевременная реакция на запросы оборудования или иных функциональных элементов	Рациональное распределение ресурсов компьютера между пользователями и задачами
Направленность	Обработка внешних событий	Обработка действий пользователя
Интерфейс	Инструмент управления аппаратно-программным комплексом	Набор приложений, готовых к использованию
Требования к пользователю	Квалифицированный разработчик	Пользователь средней квалификации

В отличие от ОС, которые установлены на любом ПК, применение ОСРВ всегда связано с внешним объектом (аппаратурой) и событиями, происходящими на нём.

Объект подконтрольный ОСРВ включает в себя датчики, регистрирующие события на этом объекте, модули ввода-вывода, преобразующие показатели датчиков в доступный, (цифровой) для системы вид.

Ещё одно отличие заключается в том, что в популярных ОС уже присутствует готовый набор при-

ложений, в свою очередь ОСРВ содержит минимальный набор системных приложений (ядро, системные модули, драйверы и т.д.). Таким образом, базис ОС сводится к минимуму (планировщик и примитивы синхронизации), а вся остальная функциональность выносится на другой уровень и реализуется через потоки и задачи, которые отвечают за системные вызовы. Такая минимизация стала возможной благодаря тому, что в настоящее время в программно-аппаратных комплексах обслуживаемых ОСРВ применяется архитектура клиент-сервер.

Основным принципом такой архитектуры является вынесение сервисов ОС в виде серверов на уровень пользователя, а микроядро выполняет функции диспетчера сообщений между клиентскими пользовательскими программами и серверами – системными сервисами (рис.1).

Архитектура клиент-сервер даёт возможность: повысить надежность ОС (т.к. каждый сервис является, по сути, самостоятельным приложением и его легче отладить и отследить ошибки); производить динамическую загрузку и отгрузку модулей; лучше масштабировать (поскольку ненужные сервисы могут быть исключены из системы без ущерба к ее работоспособности); повысить отказоустойчивость системы (т.к. «зависший» сервис может быть перезапущен без перезагрузки системы). При такой архитектуре приложения являются клиентами, которые запрашивают сервисы через системные вызовы.

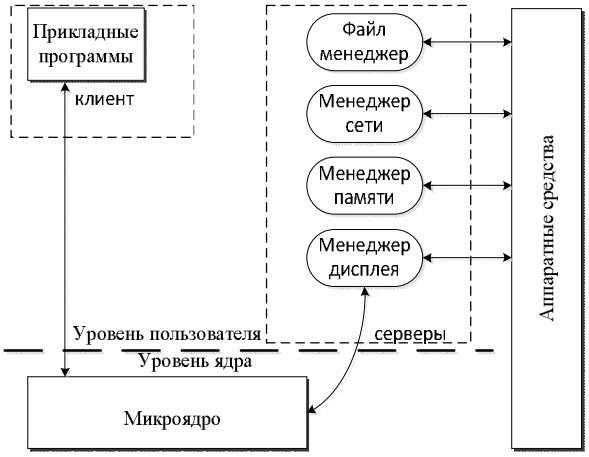

Рис. 1. Построение ОС
с использованием архитектуры клиент-сервер

Среди известных ОСРВ реализующих архитектуру клиент-сервер можно отметить OS9,VxWorks, pSOS и QNX.

Главной проблемой в этой модели архитектуры является защита памяти, поскольку серверные процессы должны быть защищены. При каждом запросе сервиса система должна переключаться с контекста приложения на контекст сервера. При поддержке защиты памяти время переключения с одного процесса на другой увеличивается.

Как правило, большинство современных ОСРВ построено на основе микроядра (kernel или nucleus), которое обеспечивает планирование и диспетчеризацию задач, а также осуществляет их взаимодействие.

По требованиям к стабильности работы различают ОС жёсткого и мягкого реального времени. В системах жёсткого реального времени опоздание реакции на событие может привести к отказам работы оборудования и катастрофическим последствиям, даже к человеческим жертвам. Системы же мягкого реального времени не так критичны к опозданиям реакции на события.

Наиболее широкий класс пользователей операционных систем реального времени – разработчики комплексов реального времени, люди проектирующие системы управления и сбора данных. Проектируя и разрабатывая конкретную систему реального времени, программист всегда точно знает, какие события могут произойти на объекте, знает критические сроки обслуживания каждого из этих событий.

Так как система реального времени – это в основном аппаратно-программный комплекс, реагирующий в предсказуемые времена на непредсказуемый поток внешних событий, то соответственно:

– система должна успеть отреагировать на событие, произошедшее на объекте, в течение времени, критического для этого события (meet deadline);

– величина критического времени для каждого события определяется объектом и самим событием, и, естественно, может быть разной, но время реакции системы должно быть предсказано (вычислено) при создании системы;

– отсутствие реакции в предсказанное время считается ошибкой для систем реального времени;

– система должна успевать реагировать на одновременно происходящие события. Даже если два или больше внешних событий происходят одновременно, система должна успеть среагировать на каждое из них в течение интервалов времени, критического для этих событий [3, 6, 7].

Основным требованием к ОСРВ является не скорость срабатывания, которая зачастую указывается поставщиком такой системы, а именно гарантия своевременного срабатывания в ответ на непредсказуемые воздействия в течение предсказуемого интервала времени. Кроме того, ОСРВ должна:

– быть многозадачной и многопрограммной;

– быть многопоточной и активно использовать прерывания для диспетчеризации;

– иметь поддержку абсолютных приоритетов;

– в ОСРВ должно применяться понятие приоритетов потока;

– иметь систему наследования приоритетов (чтобы не возникала инверсия приоритетов, поток с

низким приоритетом может заблокировать поток с более высоким приоритетом);

– средства для синхронизации процессов.

На практике ОСРВ применяется чуть ли не во всех отраслях, где присутствует техника (рис. 2).

Так как ОСРВ очень часто отвечают за работу проблемно-ориентированных компьютерных систем управления (транспортной инфраструктурой, атомными, космическими и другими объектами критического применения). В связи с этим ОСРВ должна иметь высокую степень безопасности и надёжности, своевременно реагировать на внешние воздействия.

В системах реального времени предусмотрена многопользовательская защита – это не гарантированная защита от злоумышленников, а в большей мере защита одного пользователя от другого.

Рис. 2. Основные области применения ОСРВ

Многопользовательская защита – это разделение пользователей на привилегированных и не привилегированных. Для идентификации пользователей в системе служит так называемый код идентификации пользователей UIC (User Identification Code), который состоит из двух восьмеричных чисел, каждое из которых занимает один байт (диапазон от 0 до 377), где первое число – код группы, а второе – код члена группы (например, UIC [1,54]). Кроме того в многопользовательской защите обычно распознавание пользователей происходит не только по UIC-коду, а и по номеру терминала за которым работает пользователь, это обычно имеет значение если UIC-коды двух пользователей совпадают.

Необходимыми механизмами защиты от НСД являются: дискреционное и мандатное разграничение доступа, контроль и маркировка печатных документов, контроль ввода/вывода на внешние носители, и другие. Программный код данных механизмов защиты находится в исполняемых файлах и в микроядре.

Разграничение доступа субъектов к объектам, основанное на назначении метки конфиденциальности для информации, содержащейся в объектах, и выдаче официальных разрешений (допуска) субъектам на обращение к информации такого уровня конфиденциальности. Также иногда называют: «Принудительный контроль доступа». Это способ, сочетающий защиту и ограничение прав, применяемый по отношению к компьютерным процессам, данным и системным устройствам и предназначенный для предотвращения их нежелательного использования.

Очевидно, что система, которая обеспечивает разделение данных и операций в компьютере, должна быть построена таким образом, чтобы её нельзя было «обойти». Она также должна давать возможность оценивать полезность и эффективность используемых правил и быть защищённой от постороннего вмешательства [4]. Дискреционное (избирательное) управление доступом субъектов к объектам производится на основе списков управления доступом или матрицы доступа [5].

Под контролем ввода/вывода на внешние носители подразумевается ограничение действий с этими устройствами (например, разрешить только чтение).

Однако как использование многопользовательской защиты, так и разграничение доступа, не даёт

гарантированной защиты от НСД, целью которого может быть [7]:

– чтение и/или модификация данных, которые в дальнейшем будут храниться или редактироваться на компьютере;

– осуществление перехвата различной ключевой информации, используемой для защиты данных;

– использование захваченного компьютера в качестве плацдарма для захвата других компьютеров сети.

– уничтожение хранящейся на компьютере информации или выведение компьютера из строя путем запуска вредоносного программного обеспечения.

Выводы

Таким образом, в связи с применением ОСРВ в проблемно-ориентированных компьютерных системах встаёт вопрос о разработке средств обеспечения информационной и функциональной безопасности в условиях ограниченных ресурсов памяти, невысокой производительности, а также требований гарантированного времени отклика, высокого уровня готовности и наличия средств автомониторинга.

Как правило, ОСРВ используются в тех сегментах рынка, которые характеризуются повышенными требованиями к информационной и функциональной безопасности. Очень важное значение имеет стоимость ошибки в указанных сегментах рынка. Пытаясь найти способы выполнения указанных требований, эти отрасли диктуют свои требования к ОСРВ, а именно:

– ОСРВ должна обеспечивать высокую степень "живучести" системы так, чтобы при отказе какой-либо части ПО, другая часть ПО продолжала нормально функционировать, т.е. ОСРВ должна гарантировать отсутствие общего отказа системы;

– ОСРВ должна удовлетворять жёстким требованиям по качеству ПО, что подразумевает соответствие различным отраслевым, национальным и международным стандартам, т.е. ПО должно иметь доказанное качество и во многих случаях должно быть сертифицировано уполномоченными организациями;

– вероятность сбоя в ПО должна быть минимальная;

– требования по безопасности (safety) и секретности (security) данных: в системе должны быть предусмотрены средства защиты наиболее важной информации.

Список литературы

1. Бурдонов И.Б. Операционные системы реального времени / И.Б. Бурдонов, А.С. Косачев, В.Н. Пономаренко. – М.: ИСП РАН, препринт 14. –2006. – 18 с.

2. Барретт С.Ф. Встраиваемые системы. Проектирование приложений на микроконтроллерах семейства 68HC12 / HCS12 с применением языка / С.Ф. Барретт, Д.Дж Пак // С. – М.: Изд. дом «ДМК. пресс», 2007. – 640 с.

3. Избирательное управление доступом [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Избирательное_управление_доступом.

4. Мандатное управление доступом. [Электронный ресурс]. – Режим доступа: https://ru.wikipedia.org/wiki/Мандатное_управление_доступом.

5. Панасенко С.Методы и средства защиты от несанкционированного доступа. / С. Панасенко [Электронный ресурс]. – Режим доступа к ресурсу: http://www.panasenko.ru/Articles/77/77.html

6. Жданов А.А. Операционные системы реального времени. Подробная информация об организации / А.А. Жданов // ЗАО "РТСофт", PCWeek, 8/1999. [Электронный ресурс]. – Режим доступа к ресурсу: http://asutp.ru/?p=600591

7. Семенов С.Г. Безопасность операционных систем реального времени в автоматизированных системах управления технологическим процессом / С.Г. Семенов, С.Ю. Гавриленко, В.В. Давыдов // Авіаційно- космічна техніка і технологія. – 2011. – № 8(85). – С. 222-225

Поступила в редколегію 11.02.2015

Рецензент: д-р техн. наук, с.н.с. С.Г. Семенов, Национальный технический университет «ХПИ», Харьков.

АНАЛІЗ ВИМОГ ІНФОРМАЦІЙНОЇ ТА ФУНКЦІОНАЛЬНОЇ БЕЗПЕКИ ДО ОПЕРАЦІЙНИХ СИСТЕМ РЕАЛЬНОГО ЧАСУ

Т.М. Шипова, В.В. Босько, І.А. Березюк

В роботі проведено аналіз відмінних особливостей операційних систем реального часу, вимог інформаційної та функціональної безпеки до цих систем в умовах використання їх в проблемно-орієнтованих комп'ютерних комплексах. Представлено огляд основних галузей, в яких можуть бути використані операційні системи реального часу. Зроблено висновки про необхідність виконання заданих вимог до операційних систем реального часу і використання методів і засобів забезпечення інформаційної та функціональної безпеки.

Ключові слова: *операційні системи реального часу, безпека, вимоги інформаційної та функціональної безпеки.*

ANALYSIS OF THE REQUIREMENTS OF INFORMATION AND FUNCTIONALITY SAFE TO REAL TIME OPERATING SYSTEM

T.M. Shipov, V.V. Bosko, I.A. Berezyuk

The analysis of the distinctive features of real time operating systems, and functional requirements of information security to these systems in terms of their use in problem-oriented computer complexes. An overview of the main branches, which can be used real time operating system. The conclusions about the need for performing the specified requirements for the real time operating system and the use of methods and means of information and functional safety.

Keywords: *real-time operating systems, security, and functional requirements of information security.*