УДК 004.43, 004.56

О РАБОТАХ ПО СОЗДАНИЮ ДОВЕРЕННОГО ПРОГРАММНО-АППАРАТНОГО ОБЕСПЕЧЕНИЯ ДЛЯ СИНТЕЗА БЕЗОПАСНЫХ КОМПЬЮТЕРНЫХ СИСТЕМ ЛОГИЧЕСКОГО УПРАВЛЕНИЯ НА БАЗЕ РУССКОГО ЯЗЫКА ПРОГРАММИРОВАНИЯ

Г. П. Агибалов

Томский государственный университет
Россия, 634050, г. Томск, просп. Ленина, 36. E-mail: agibalov@isc.tsu.ru

Создание безопасных компьютерных систем логического управления стратегическими объектами возможно лишь на основе собственного (*доверенного*) *программно-аппаратного обеспечения. Сообщается о состоянии дел и перспективах разработки такого обеспечения на базе русского языка программирования, широко распространённого в 1960–1980-е годы в Советском Союзе и за рубежом под названием ЛЯПАС* (*LYaPAS*) *и ныне возрождаемого с этой целью в Томском государственном университете.*

Ключевые слова: *компьютерная безопасность, криптография, системы логического управления, русский язык программирования, ЛЯПАС, программно-аппаратное обеспечение*.

RESEARCH IN PRODUCTION OF TRUSTED SOFT AND HARDWARE FOR THE SYNTHESIS OF SECURE COMPUTER SYSTEMS OF LOGICAL CONTROL ON THE RUSSIAN PROGRAMMING LANGUAGE BASIS

G. P. Agibalov

Tomsk State University
36, Lenin prosp., Tomsk, 634050, Russia. E-mail: agibalov@isc.tsu.ru

The secure computer systems destined for the logical control of the strategic objects (*like space apparatuses, nuclear weapons, energetic stations, transport means, submarines, etc*.) *can not be designed on the base of the untrusted soft and hardware produced by a potential adversary. The information is given about the results and perspectives of research in production to create such systems of the soft and hardware based on the Russian programming language LYaPAS which used to be widely known and applied in USSR and abroad during 1960–1980 and which is successfully reanimated now at Tomsk state university.*

Keywords: *computer security, cryptography, logical control, Russian programming language LYaPAS, software, hardware*.

В настоящее время наиболее серьёзные угрозы информационной безопасности нашей страны (как внутри, так и извне) проистекают от использования в компьютерных системах (КС) управления критически важными объектами (космическими системами, транспортными средствами, ядерными установками, подводными лодками, беспилотниками и т. п.) недоверенного программно-аппаратного обеспечения, заимствованного у своего же потенциального противника. Оно, это обеспечение, как правило, несёт в себе недокументированные закладки, через которые возможна утечка одной информации и навязывание другой, в том числе разрушительной, и обнаружить которые, даже в свободном коде, не всегда возможно. Предотвращение подобных угроз для России возможно лишь путём использования в КС собственного программно-аппаратного обеспечения на базе русского языка программирования. Да, такой язык есть. Он создан в начале 1960-х годов в Томском государственном университете под руководством Аркадия Дмитриевича Закревского для решения задач автоматического синтеза систем логического управления. Русским его назвали американцы – Russian programming language, а в оригинале он называется ЛЯПАС (с ударением на второй слог) – логический язык для представления алгоритмов синтеза. С его богатой историей до 1990-х годов можно познакомиться по [1].

Позднее, на волне демократии, с ликвидацией военно-промышленного комплекса и производства отечественных ЭВМ, интерес к ЛЯПАСу пропал. Сейчас же, когда осознание необходимости для национальной безопасности собственного доверенного программно-аппаратного обеспечения КС возвращается, стала актуальной проблема возрождения ЛЯПАСа [2] с целью создания на его базе такого обеспечения для синтеза безопасных КС логического управления и криптографической защиты управляющей информации. Достоинства целевого продукта, обусловленные свойствами ЛЯПАСа, следующие:

– на все 100 % доверенный отечественный продукт с открытым и легко читаемым кодом;

– язык программирования, гарантирующий написание программ без уязвимостей;

– безопасная операционная система (ОС), исключающая возможность запуска злонамеренного кода;

– библиотека прикладных программ с реальным временем исполнения;

– реактивная ОС, гарантирующая исполнение алгоритмов управления в реальном времени;

– гарантированное отсутствие недокументированных закладок в КС, создаваемых с его помощью;

– возможность создания специализированного компьютера (ЛЯПАС-машины) с процессором, аппаратно реализующим язык программирования.

Это будет недорогой одноадресный компьютер с системой команд, образующих исполняемый код ЛЯПАСа, в котором операции и операнды последнего представлены своими кодами и адресами в памяти компьютера. Скорость исполнения программ в нём на несколько порядков выше скорости их исполнения в существующих компьютерах. Столь же простая и эффективная аппаратная реализация других языков программирования вряд ли возможна ввиду их принципиально иного устройства. Попытки аппаратной реализации языка Java, например, по большому счёту не увенчались успехом.

Вместе с хранилищем ЛЯПАСных программ ЛЯПАС-машина может стать простейшей и вместе с тем эффективнейшей универсальной безопасной компьютерной системой управления любыми сложными объектами. Для её применения в управлении конкретным объектом достаточно будет загрузить в это хранилище исполняемый код программы на ЛЯПАСе, реализующей алгоритм управления данным объектом.

Аналогичным образом ЛЯПАС-машину можно использовать и в роли криптопроцессора, предназначенного для эффективного исполнения криптографических алгоритмов.

Состояние дела по возрождению ЛЯПАСа на данный момент следующее.

1. Произведена ревизия ЛЯПАСа, результатом которой стал язык vЛЯПАС (reVised LYaPAS) с алфавитом операций и операндов, приспособленным к современным средствам отображения информации (используется символика Unicode и TEX) [3; 4].

2. Разработано криптографическое расширение vЛЯПАСа – язык ЛЯПАС-Т (от exTended LYaPAS), вобравший в себя в качестве элементарных операции из современных криптографических алгоритмов [3; 4].

3. Создан и запущен в опытную эксплуатацию компилятор с vЛЯПАСа в язык «Ассемблер» под ОС Linux. С его помощью ведётся отладка, исполнение, экспериментальное исследование алгоритмов на vЛЯПАСе и их доработка по результатам исследования [3; 4].

4. Разработаны проекты процессора, аппаратно реализующего ЛЯПАС-Т, и препроцессора, транслирующего программы на ЛЯПАС-Т в его исполняемый код [4; 5].

5. Разработана и описана на VHDL архитектура процессора vЛЯПАСа 1-го уровня, построена программируемая логическая интегральная схема (ПЛИС) этого процессора.

6. Создан ряд прикладных программ на vЛЯПАСе для криптографической защиты управляющей информации.

Дальнейшие исследования предполагают решение следующих задач.

1. Разработка и исследование математического и программного обеспечения:

– математической модели безопасной компьютерной системы логического управления;

– средств трансляции и отладки программ на ЛЯПАС-Т для их исполнения на современных компьютерах;

– алгоритмов и программ на ЛЯПАС-Т для логического управления, логического синтеза управляющих автоматов и криптографической защиты управляющей информации;

– реактивной ОС на vЛЯПАСе для управления в реальном времени.

2. Исследование и разработка аппаратного обеспечения:

– процессора ЛЯПАС-Т (его исполняемого кода, архитектуры, алгоритма функционирования);

– препроцессора, транслирующего программы на ЛЯПАС-Т в исполняемый код процессора;

– архитектуры ЛЯПАС-машины;

– ОС, управляющей взаимодействием процессора ЛЯПАС-машины с аппаратными модулями системы управления.

3. Реализация процессора ЛЯПАС-Т на базе ПЛИС и (или) заказных интегральных схем:

– описание процессора на VHDL;

– его отладка путём компьютерного моделирования по компонентам и в целом;

– автоматический синтез и компьютерное моделирование логической схемы процессора.

Библиографические ссылки

1. Торопов Н. Р. Язык программирования ЛЯПАС // Прикладная дискретная математика. 2009. № 2 (4). С. 9–25.

2. Агибалов Г. П. К возрождению русского языка программирования // Прикладная дискретная математика. 2012. № 3 (17). С. 77–84.

3. Agibalov G. P., Lipsky V. B., Pankratova I. A. Cryptographic extension of Russian programming language // Прикладная дискретная математика. Приложение. 2013. № 6. С. 93–98.

4. Агибалов Г. П., Липский В. Б., Панкратова И. А. О криптографическом расширении и его реализации для русского языка программирования // Прикладная дискретная математика. 2013. № 3 (21). С. 93–104.

5. Agibalov G. P., Lipsky V. B., Pankratova I. A. Project of hardware implementation of Russian programming language // Прикладная дискретная математика. Приложение. 2013. № 6. С. 98–102.

References

1. Toropov N. R. *Prikladnaya diskretnaya matematika*. 2009. № 2 (4). pp. 9–25.

2. Agibalov G. P. *Prikladnaya diskretnaya matematika*. 2012. № 3 (17). pp. 77–84.

3. Agibalov G. P., Lipsky V. B., Pankratova I. A. Cryptographic extension of Russian programming language. *Prikladnaya diskretnaya matematika*. *Prilozhenie*. 2013. № 6. pp. 93–98.

4. Agibalov G. P., Lipsky V. B., Pankratova I. A. *Prikladnaya diskretnaya matematika*. 2013. № 3 (21). pp. 93–104.

5. Agibalov G. P., Lipsky V. B., Pankratova I. A. Project of hardware implementation of Russian programming language. *Prikladnaya diskretnaya matematika*. *Prilozhenie*. 2013. № 6, pp. 98–102.

© Агибалов Г. П., 2013

УДК 004.7.056.53

СИСТЕМА АКТИВНОГО МОНИТОРИНГА СВОЙСТВ БЕЗОПАСНОСТИ СЕТЕВЫХ УЗЛОВ

Д. А. Бородавкин[1], *И. В. Потуремский*[2], *Д. П. Маренков*[3]

ОАО «Информационные спутниковые системы» имени академика М. Ф. Решетнева»
Россия, 662972, г. Железногорск Красноярского края, ул. Ленина, 52
E-mail: db@iss-reshetnev.ru[1], oris@iss-reshetnev.ru[2], marenkovd@yandex.com[3]

Рассматриваются принципы построения системы активного мониторинга свойств безопасности сетевых узлов и некоторые практические аспекты ее реализации. Описанная система решает задачи автоматизированного мониторинга сетевых узлов и сервисов, контроля уязвимостей и оповещения о выявлении соответствующих событий информационной безопасности. Система функционирует, производя анализ статистики о потоках данных в сети, и при определенных условиях инициирует активное сканирование сервисов.

Ключевые слова: сетевые технологии, информационная безопасность, контроль уязвимостей.

ACTIVE SECURITY MONITORING SYSTEM OF NETWORK DEVICES

D. A. Borodavkin[1], *I. V. Potyremskiy*[2], *D. P. Marenkov*[3]

JSC “Academician M. F. Reshetnev “Information Satellite Systems”
52, Lenin str., Zheleznogorsk, Krasnoyarsk region, 662972, Russia
E-mail: db@iss-reshetnev.ru[1], oris@iss-reshetnev.ru[2], marenkovd@yandex.com[3]

The principles of network devices active security monitoring system building and some practical development issues are described. The aim of the system is to solve the problems of network devices and service automated monitoring, vulnerabilities control and information security events notification. The system acts performing network data stream analysis and initiate active service scan under meeting some conditions.

Keywords: computer networking, information security, vulnerability control.

Работа пассивных систем обеспечения информационной безопасности (ИБ) должна сопровождаться непрерывным контролем состояния ИБ со стороны автоматизированных систем аудита [1]. Кроме того, задачи обеспечения ИБ предполагают некоторую автономность от информационной инфраструктуры предприятия. Ниже излагается схема работы системы активного мониторинга сетевых узлов, позволяющей вести непрерывный мониторинг узлов и сервисов в сети.

В целом, система решает задачи контроля подключения устройств к сети предприятия, мониторинга сетевых сервисов на узлах сети и уязвимостей этих сервисов. На практике такой функционал может использоваться:

– для контроля уязвимостей демилитаризованной зоны;

– контроля подключений клиентов локальной сети;

– аудита использования запрещенных сервисов локальной сети;

– выделения деятельности вредоносного ПО типа Backdoor.

Принцип работы системы заключается в следующем:

1. На основании информации о потоках данных в сети, получаемой от коммутационного оборудования (протокол NetFlow [2]), непрерывно формируется список узлов сети и их активных сетевых портов.

2. Для узлов из списка осуществляется сканирование портов с целью определения состава сервисов, доступных на хосте. Сканирование производится по расписанию, а также по срабатыванию определенных триггеров.

3. Данные сканирований систематизируются, выделяются и ранжируются события ИБ (обнаружение нового узла, обнаружение нелегитимного узла, обна-