ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

ПРЕДВАРИТЕЛЬНЫЙ НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ПНСТ 818— 2023

Информационные технологии

ИНТЕРНЕТ ВЕЩЕЙ

Системы с разделением доменов. Базовые компоненты

Издание официальное

Москва
Российский институт стандартизации
2023

Предисловие

1 РАЗРАБОТАН Акционерным обществом «Лаборатория Касперского» (АО «Лаборатория Касперского»)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 194 «Кибер-физические системы»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 13 марта 2023 г. № 13-пнст

Правила применения настоящего стандарта и проведения его мониторинга установлены в ГОСТ Р 1.16—2011 (разделы 5 и 6).

Федеральное агентство по техническому регулированию и метрологии собирает сведения о практическом применении настоящего стандарта. Данные сведения, а также замечания и предложения по содержанию стандарта можно направить не позднее чем за 4 мес до истечения срока его действия разработчику настоящего стандарта по адресу: 121205 Москва, Инновационный центр Сколково, улица Нобеля, 1, e-mail: info@tc194.ru и/или в Федеральное агентство по техническому регулированию и метрологии по адресу: 123112 Москва, Пресненская набережная, д. 10, стр. 2.

В случае отмены настоящего стандарта соответствующая информация будет опубликована в ежемесячном информационном указателе «Национальные стандарты» и также будет размещена на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (www.rst.gov.ru)

Содержание

Введение

Основной идеей систем с разделением доменов является создание конструктивного решения, устойчивого к реализации угроз информационной безопасности и к некоторым другим видам опасностей, действующих в отношении Интернета вещей.

Описание архитектурного подхода с разделением доменов и, в частности, базовых компонентов систем с разделением доменов закладывает основу для описания частных решений, реализующих различные функции безопасности и средства защиты от компьютерных угроз.

Целью описания базовых компонентов систем с разделением доменов не является описание конкретных, поименованных функций и механизмов безопасности. Функции и механизмы безопасности сами по себе могут быть реализованы уязвимым образом и представлять опасность для якобы защищенной с их помощью системы.

Одной из задач предлагаемой серии стандартов является описание методических подходов к проектированию систем и включаемых в их состав необходимых механизмов безопасности, а также типовых архитектурных решений (шаблонов) внедрения функций безопасности с тем, чтобы обеспечить доверие к этим механизмам и функциям.

ПРЕДВАРИТЕЛЬНЫЙ НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Информационные технологии

ИНТЕРНЕТ ВЕЩЕЙ

Системы с разделением доменов. Базовые компоненты

Information technology. Internet of things. Domain shared systems. Base components

Срок действия — с 2023—03—31
по 2026—03—31

1 Область применения

Настоящий стандарт определяет перечень основных базовых компонентов систем Интернета вещей, использующих подход с разделением доменов.

Также в стандарте предоставлены рекомендации по использованию указанных компонентов в системах с разделением доменов.

Настоящий стандарт предназначен для применения при проектировании и эксплуатации систем Интернета вещей, использующих подход с разделением доменов.

2 Нормативные ссылки

В настоящем стандарте использована нормативная ссылка на следующий стандарт:

ПНСТ 819—2023 Информационные технологии. Интернет вещей. Системы с разделением доменов. Термины и определения

Примечание — При пользовании настоящим стандартом целесообразно проверить действие ссылочных стандартов в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет или по ежегодному информационному указателю «Национальные стандарты», который опубликован по состоянию на 1 января текущего года, и по выпускам ежемесячного информационного указателя «Национальные стандарты» за текущий год. Если заменен ссылочный стандарт, на который дана недатированная ссылка, то рекомендуется использовать действующую версию этого стандарта с учетом всех внесенных в данную версию изменений. Если заменен ссылочный стандарт, на который дана датированная ссылка, то рекомендуется использовать версию этого стандарта с указанным выше годом утверждения (принятия). Если после утверждения настоящего стандарта в ссылочный стандарт, на который дана датированная ссылка, внесено изменение, затрагивающее положение, на которое дана ссылка, то это положение рекомендуется применять без учета данного изменения. Если ссылочный стандарт отменен без замены, то положение, в котором дана ссылка на него, рекомендуется применять в части, не затрагивающей эту ссылку.

3 Термины и определения

В настоящем стандарте применены термины по ПНСТ 819—2023.

4 Системы с разделением доменов. Базовые компоненты

4.1 Общие положения

Создание систем Интернета вещей с учетом необходимости обеспечения их информационной безопасности и функциональной безопасности требует принимать во внимание особенности таких си-

Издание официальное

стем. Проектирование систем Интернета вещей не всегда можно рассматривать как частный случай общей дисциплины проектирования информационных систем. Системы Интернета вещей функционируют как в информационном, так и в физическом окружении, следовательно, аспекты их безопасности разнообразнее и сложнее в своем сочетании, чем аспекты безопасности информационных систем.

Принципы проектирования и требования к разработке автоматизированных систем управления технологическим процессом далеко не в полной мере распространяются на системы Интернета вещей. Отсутствуют рекомендованные методы обеспечения безопасности, подходящие для систем Интернета вещей. Это приводит к повышенной подверженности систем Интернета вещей угрозам информационной безопасности (по сравнению с информационными системами). В свою очередь, угрозы информационной безопасности системам Интернета вещей могут приводить к последствиям, связанным с функционированием систем в физическом окружении. Это влечет необходимость рассмотрения вопросов функциональной безопасности этих систем одновременно с вопросами информационной безопасности.

Архитектурный подход с разделением доменов — это стратегия экономически эффективного построения систем Интернета вещей, требующих функциональной и информационной безопасности с высокой степенью уверенности. Это компонентный подход к проектированию, реализации и сертификации безопасных систем Интернета вещей, основанный на реализации доверенной операционной системы.

При проектировании и реализации систем с разделением доменов особое внимание уделяется декомпозиции, разделению и взаимодействию ее компонентов, а также безопасному совместному использованию вычислительных ресурсов. Классическими подходами к декомпозиции программных систем является вертикальная (иерархическая) и горизонтальная (компонентная) декомпозиция. Вследствие использования вертикальной и горизонтальной декомпозиции задачи проектирования, верификации, и сертификации могут выполняться на уровне операционной системы, уровне приложений и уровне системы в целом.

Вертикальная и горизонтальная декомпозиция структуры системы с разделением доменов представлена на рисунке 1.

На рисунке в соответствии с вертикальной декомпозицией есть три основных типа компонентов:

- ядро разделения,
- промежуточное программное обеспечение (ПО),
- прикладные программы.

Как следует из рисунка, модули промежуточного ПО могут использовать функции модулей ядра разделения, а прикладные программы могут использовать функции модулей ядра разделения и промежуточного ПО.

Требования к аппаратному обеспечению для системы на основе разделения доменов зависят от политики разделения доменов и внешних интерфейсов, определяемых назначением этой системы и ее функциональностью.

Большинство современных операционных систем (ОС) поддерживают двухуровневую систему привилегий: режим ядра (пространство ядра) и режим пользователя (пространство пользователя). В режиме ядра выполняются все разрешенные инструкции, в ходе выполнения доступна вся оперативная память и аппаратные ресурсы.

Ядро разделения работает в режиме ядра, промежуточное ПО и прикладные программы работают в пользовательском режиме.

Наиболее часто ядро разделения реализуется:

- как микроядро операционной системы;
- гипервизор виртуализации (менеджер виртуальных машин) системы.

В пользовательском режиме доступ к памяти и аппаратным ресурсам ограничен. Прикладной программе не будет позволено работать с памятью за пределами набора адресов, установленной ОС, или обращаться напрямую к аппаратным ресурсам. Поддержка двухуровневой системы привилегий должна осуществляться аппаратным обеспечением, на уровне процессора.

Базовые компоненты ядра разделения, определяющие его функциональные возможности, включают механизмы:

- управления памятью и разделения ресурсов на домены,
- поддержки междоменной связи,
- проверки выполнения политики безопасности,
- планирования,

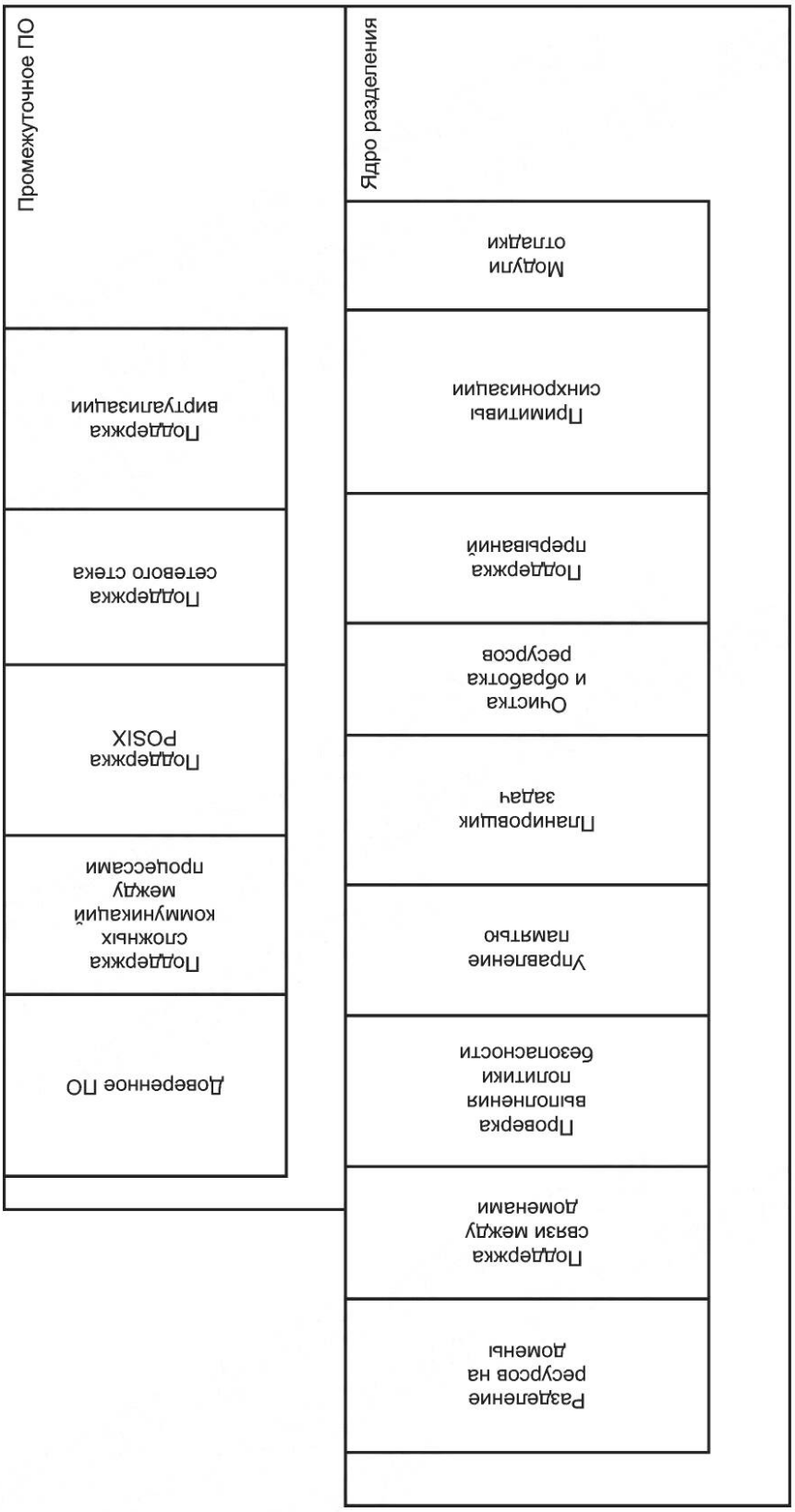

Рисунок 1 — Компонентная структура операционной системы с разделением доменов

- очистки и обработки ресурсов между периодами их раздельного использования,
- минимального обслуживания прерываний,
- управления минимальными примитивами синхронизации, таймеры и сторожевые таймеры,
- управления контрольно-измерительными приборами (при необходимости).

Основным требованием к модулям ядра является то, что они должны быть компактными и простыми, чтобы позволить выполнить формальную проверку их правильности за конечное время.

4.2 Ядро разделения

Ядро ОС включает компоненты, реализующие функции и структуры данных, представляющие наиболее низкий уровень абстракции для доступа прикладных программ к ресурсам системы.

Ядро безопасности, наследуя компонентный набор, определяемый ядром ОС, включает компонент монитора пересылок (см. ПНСТ 819—2023), реализуемый на основе комбинации аппаратного и программного обеспечения.

Ядра разделения предоставляют набор функций для обеспечения разделения доменов. Ядро разделения является специальным типом ядер безопасности, которые в свою очередь являются специальным типом ядер ОС (см. рисунок 2).

Рисунок 2 — Иерархия ядер ОС (фрагмент)

Отличительным признаком системы с разделением доменов является разделение ресурсов на домены. Ядро разделения должно поддерживать разделение ресурсов на всех уровнях системы, от оборудования до прикладных программ. Хотя функции разделения доменов могут быть реализованы различным образом, поддержка разделения на возможно более низком уровне упрощает реализацию и верификацию поведения системы (при условии доверия ее аппаратному обеспечению). Разделение на домены выполняется при проектировании системы, с учетом независимости доменов и их функциональной изоляции.

Ядро разделения реализует пространственное и временное разделение доменов, что позволяет организовать контролируемое взаимодействие доменов. Пространственное разделение доменов поддерживает разделение данных, управление информационным потоком и изоляцию отказов. Разделение данных означает, что каждый домен развернут как отдельный ресурс. Приложения в одном домене не могут ни косвенно влиять на данные в других доменах, ни контролировать приложения и устройства в них. Управление информационным потоком обеспечивает соответствующий политике безопасности поток информации между доменами. Изоляция отказов ограничивает распространение отказов от одного домена к другому.

Реализация функций пространственного разделения достигается за счет использования общих аппаратных устройств, таких как модуль управления памятью (MMU) и модуль управления памятью ввода / вывода (IOMMU).

При верификации ядра разделения следует выявлять скрытые каналы, которые могут образоваться при разделении доменов. Скрытый канал — это канал передачи информации, который не был предусмотрен для связи доменов. Могут возникнуть скрытые каналы, связанные с ресурсами, и скрытые каналы, связанные с планированием задач (см. 4.5).

Временное разделение доменов поддерживает планирование задач во времени, меры по предотвращению временной задержки доступа к ресурсам, совместно используемым доменами с течением времени. Кроме того, временное разделение доменов обеспечивает безопасную очистку ресурсов. Безопасная очистка ресурсов обеспечивает отсутствие несанкционированного доступа к данным одного при переключении к обработке данных других доменов. Безопасная очистка ресурсов помогает устранить скрытые каналы и защититься от атаки «холодной загрузки» и других атак на конфиденциальные данные внутри доменов, основанных на повторном использовании ресурсов.

Разделение на домены должно быть таким, что при оценке системы можно было доказать, что ни одно приложение в домене не оказывает отрицательного влияния на поведение приложения в другом домене.

Ядро разделения определяет авторизованные каналы между доменами для обеспечения междоменной связи. Доступ к данным может быть осуществлен только по этим каналам, реализующим принцип отказа по умолчанию.

4.3 Поддержка междоменной связи

Поддержка междоменной связи реализуется ядром ОС. Различные системы и платформы с разделением доменов реализуют междоменную связь с использованием разных транспортных механизмов. К таким механизмам можно отнести:

а) передачу сообщений:
- сокет (архитектура POSIX),
- канал (именованный, неименованный) (архитектура POSIX),
- очередь сообщений (архитектура POSIX),
- почтовый ящик;

б) механизмы синхронизации:
- семафор (архитектура POSIX),
- сигнал (архитектура POSIX);

в) механизмы разделения памяти:
- разделяемая память (архитектура POSIX),
- проецируемый в память файл (архитектура POSIX).

Набор механизмов междоменной связи может быть реализован в зависимости от требований к функциональности ядра разделения. Единственное требование, предъявляемое подходом с разделением доменов к междоменной связи, — это поддержка выделенных однонаправленных каналов для обмена информацией.

Ядро разделения определяет авторизованные каналы между доменами для обеспечения междоменной связи. Временные требования устанавливают требования к необходимой пропускной способности ядра разделения. Производительность обмена информацией ограничена необходимостью проверять соответствие политике безопасности для междоменной связи. Поэтому при проектировании ядра разделения необходимо найти баланс между производительностью и безопасностью.

Ограничения на реализацию механизмов междоменной связи могут быть сформулированы на основе ограничений на сложность ПО. Сложность реализации ядра разделения должна быть ограничена для обеспечения возможности проверки, верификации и сертификации. По этой причине, например, реализация синхронной междоменной связи в ядре предпочтительнее, поскольку асинхронные механизмы связи требуют выделения буферов и реализации примитивов синхронизации (см. 4.6). Для поддержки эффективного взаимодействия между приложениями в различных доменах платформа с ядром разделения должна реализовывать протоколы междоменной связи высокого уровня или даже стек протоколов поверх примитивов транспортных механизмов, реализованных разделительным ядром (см. 4.7).

4.4 Проверка выполнения политики безопасности

Проверка выполнения политики безопасности основана на реализации монитора пересылок. Монитор пересылок обрабатывает каждый запрос на доступ к данным и принимает решение о доступе в

соответствии с определенной политикой безопасности. То есть монитор пересылок должен проверять все междоменные взаимодействия на соответствие политике безопасности. Корректная реализация монитора пересылок требует обмена параметрами безопасности процесса домена с монитором пересылок, которые являются частью содержимого сообщения. Это требует реализации защищенного от несанкционированного доступа механизма передачи решения о безопасности обратно в ядро разделения. Реализация такого механизма повышает сложность ядра разделения.

4.5 Управление памятью и планирование. Периоды и прерывания

Разделение данных требует, чтобы адресные пространства памяти каждого домена были независимы от других. Это требование может быть реализовано с использованием аппаратных функций управления памятью, предоставляемых ядром разделения.

Методы управления памятью (такие, как перераспределение памяти) должны быть ограничены, чтобы их можно было проверить и сделать выводы о поведении системы на основе разделения доменов. Например, должна быть верифицирована корректная реализация требования по очистке использованной памяти (см. 4.2).

Планирование выполнения задач заключается в назначении приоритетов процессам в очереди с приоритетами. Программный код, выполняющий эту задачу, называется планировщиком. Алгоритмы, используемые планировщиком, определяются требованиями к временному разделению (см. 4.2) и возможными требованиями при выполнении в реальном времени. Планирование выполнения задач может привести к появлению скрытых каналов синхронизации между доменами.

Реализация очистки ресурсов также может повлиять на предсказуемость поведения системы. Синхронизация скрытых каналов может возникнуть, например, когда задержки и временные интервалы предопределены для конкретных действий или событий и делают возможным скрытые каналы взаимодействия. Реализация очистки ресурсов может породить скрытые каналы взаимодействия и требует проверки недетерминированных временных задержек. Такую проверку целесообразно проводить в период оценки доверия системы.

Реализация ядра разделения требует особого внимания к переключению между доменами, когда ресурс, выделенный одному компоненту в течение одного периода, очищается, после чего выделяется другому компоненту.

Прерывание — сигнал процессору, устанавливаемый аппаратными средствами или программным обеспечением и указывающий на событие, которое требует немедленного реагирования. Прерывание может произойти в любой момент времени, поэтому обработчик прерывания может выполняться в любое время и в любом контексте. Обработчик, если он находится в ядре, должен быстро выполнить процедуру обработки и возобновить выполнение прерванного кода как можно скорее. Для ядер разделения важно, чтобы обработка прерываний могла выполняться в драйверах пространства пользователя. При этом в ядре все равно останется небольшой фрагмент кода, обслуживающий контроллер прерываний, но ядро обязано уметь доставлять сигнал о прерывании в пользовательский драйвер, а также должен быть механизм подтверждения обработки прерывания этим драйвером. Для систем реального времени время обработки прерывания должно быть детерминированным.

4.6 Примитивы синхронизации

Методы синхронизации необходимы тогда, когда существует несколько доменов, которые выполняются одновременно и могут потенциально взаимодействовать друг с другом. Ядро разделения должно реализовывать примитивы синхронизации и обрабатывать аппаратные ловушки, такие как таймеры и сторожевые таймеры. К примитивам синхронизации относятся: семафоры, взаимные исключения, циклические блокировки. Требования к управлению таймером определяются ограничениями планирования.

4.7 Промежуточное ПО

Ядро разделения лежит в основе архитектуры системы на основе разделения доменов, реализуя основные примитивы для междоменной связи, предоставляя остальным процессам в системе соответствующие интерфейсы. Пользовательские процессы, работающие в системе, могут использовать расширенные сервисы и протоколы, такие как асинхронная связь с отслеживанием состояния соединения. Такие сервисы сложны и не могут быть частью ядра разделения в связи с тем, что их трудно формально верифицировать. Поэтому их целесообразно реализовать как промежуточное ПО.

Промежуточное программное обеспечение может поддерживать стандартизированные интерфейсы для приложений, требующих совместимости, например, POSIX-совместимости.

Промежуточное программное обеспечение также может реализовывать уровень управления виртуализацией (управление и контроль виртуальных машин, представляющих домены, ограничены принципами системы с разделением доменов и архитектурой политик) и уровень совместимости, который может запускать операционные системы в домене.

Одной из технологий, поддерживающих разделение доменов, является виртуализация. Виртуализация, поддерживаемая процессором, концептуально вводит новый уровень привилегий, предоставляя операционной системе косвенный доступ к оборудованию. В этом случае операционная система связывается с гипервизором, который реализует функциональные возможности ядра разделения, включая управление оборудованием и очистку и обработку ресурсов между периодами их раздельного использования. Это позволяет запускать несколько, возможно, разных операционных систем на одном процессоре и распределять аппаратные ресурсы между ними. При использовании технологии виртуализации выделение аппаратных ресурсов каждому домену должно быть корректно реализовано с учетом безопасной очистки.

Для многоядерных процессоров гипервизор отвечает за создание виртуализированной среды для каждого ядра, настройку защиты памяти, необходимой для каждого домена, виртуализацию устройств, а также за загрузку и запуск соответствующего программного обеспечения.

Помимо уровня промежуточного программного обеспечения система с разделением доменов может поддерживать дополнительные сервисы, которые реализуют функции, связанные с безопасностью, и составляют часть доверенного ПО системы. Они могут включать идентификацию и аутентификацию пользователя, контроль целостности данных, доменов и приложений, мониторинг, аудит безопасности, функции криптографической защиты, доверенную загрузку, проверку целостности кода ОС, безопасное хранилище и другие функции, связанные с безопасностью.

Эти сервисы не реализуются самим ядром разделения из-за требования простоты и минимизации объема кода.

УДК 004.738:006.354 ОКС 35.110

Ключевые слова: Интернет вещей, системы с разделением доменов, базовые компоненты

Редактор *В.Н. Шмельков*
Технический редактор *И.Е. Черепкова*
Корректор *М.И. Першина*
Компьютерная верстка *Л.А. Круговой*

Сдано в набор 15.03.2023. Подписано в печать 16.03.2023. Формат 60×84⅛. Гарнитура Ариал.
Усл. печ. л. 1,40. Уч.-изд. л. 1,26.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Создано в единичном исполнении в ФГБУ «Институт стандартизации»
для комплектования Федерального информационного фонда стандартов,
117418 Москва, Нахимовский пр-т, д. 31, к. 2.
www.gostinfo.ru info@gostinfo.ru