УДК 681.3

К вопросу информационной безопасности Grid-систем

Шелестов А.Ю., Скакун С.В., Куссуль О.М., Хоанг Динь Хыу
Институт космических исследований НАНУ-НКАУ
inform@ikd.kiev.ua

Abstract

Shelestov A., Skakun S., Kussul O. Hoang D.H. On Enabling Security in Grid Systems. *We review different approaches to enabling security in Grid systems. We analyze such mechanisms as Globus GSI and message-level security. We also focus on the following aspects of Grid security: anomaly identification, and resource and data availability monitoring. Proposed model of user behaviour for anomaly identification is based on the analysis of statistical data about jobs that were carried out by the user in Grid system. In order to distinguish normal and abnormal user's behaviour we use feed-forward neural networks. We verified the proposed approach on data that were collected by GridICE monitoring system in GILDA-EGEE. The results of experiments showed that our model was able to distinguish abnormal user's behaviour in 90%. Resource availability monitoring is based on the concept of active experiment. This enables the analysis of availability of Grid resource in real-time.*

Введение

Технология Grid предполагает использование программного обеспечения (ПО) среднего уровня (middleware), предназначенного для объединения распределенных информационных и вычислительных ресурсов различных административных доменов в рамках единой виртуальной организации (ВО) [1–6]. Очевидно, что одной из важнейших задач при разработке подобных сложных систем является реализация механизмов обеспечения безопасности [7]. Многие вопросы обеспечения безопасности решаются локально на уровне отдельных Grid-ресурсов при использовании систем выявления вторжений и средств администрирования ресурсов. В частности, к числу таких задач относятся обеспечение аутентификации и авторизации, обмен сертификатами, обеспечение конфиденциальности и целостности данных, а также аудит и мониторинг ресурсов и пользователей [7, 8, 9]. В настоящее время большинство этих задач при построении Grid-систем решается на основе инфраструктуры Grid Security Infrastructure (GSI) [10], которая по существу представляет собой расширение инфраструктуры открытого ключа (PKI — Public Key Infrastructure) [11, 12, 13]. GSI-инфраструктура поддерживает одноразовую регистрацию (single sign on), делегирование полномочий и обмен сертификатами.

При включении отдельных ресурсов или сегментов в Grid-систему необходимо обеспечить безопасность ее функционирования на системном уровне. Одним из наиболее важных аспектов обеспечения безопасности является мониторинг действий пользователей при работе с удаленными ресурсами Grid-системы, поскольку согласно [14] 80-90% атак инициируется именно пользователями внутри компьютерной системы. Поэтому необходимо обеспечить выявление аномалий не только на уровне отдельных ресурсов и сегментов, но и всей системы в целом. Штатными средствами GSI эта задача не решается.

В настоящее время существует достаточно много средств мониторинга состояния ресурсов Grid-системы и запускаемых задач (например, GridICE [15] и MOGAS [16]). Однако эти средства не предоставляют средств анализа работы пользователей для выявления их аномальной деятельности. Поэтому на сегодняшний день разработка методов и моделей анализа поведения пользователей в сложных распределенных Grid-системах является актуальной задачей.

Заслуживает пристального внимания еще один аспект безопасности функционирования Grid-систем, обусловленный их архитектурой и не обеспечиваемый стандартным ПО. В основу таких систем положена концепция сервисов, или компонентов, взаимодействующих по сети. Одним из важнейших показателей их успешного функционирования является уровень доступности, т.е. возможность в приемлемое время обработать весь поток поступающих от пользователей заявок. Причем уровень доступности в заданный момент времени может определяться как текущей загруженностью ресурсов Grid-системы или сетевой среды, так и фактом реализации атак DoS или DDoS, направленных на генерацию условия отказа в обслуживании.

В данной статье проанализированы разные средства обеспечения безопасности в Grid-системах, а также рассматривается метод выявления аномальной деятельности пользователей и информационная технология определения доступности ресурсов в Grid-системе.

Инфраструктура защиты Globus GSI

В настоящее время ПО среднего уровня Globus Toolkit является де-факто стандартом при разработке Grid-систем. Для решения задач обеспечения безопасности современные Grid-системы используют инфраструктуру Globus Grid Security Infrastructure (GSI) [10]. Базирующийся на технологии открытых ключей протокол GSI осуществляет аутентификацию пользователя в условиях однократной регистрации, коммуникационную защиту и начальную поддержку ограниченного делегирования полномочий.

Однократная регистрация обеспечивает пользователю возможность только один раз пройти процедуру аутентификации и, таким образом, создать прокси-сертификат (proxy certificate), который может быть предъявлен программой любой удалённой службе для аутентификации от имени пользователя. Делегирование делает возможным создание и передачу удаленной службе делегированного прокси-сертификата, который может быть использован этой службой для выполнения действий от имени пользователя (возможно, с некоторыми ограничениями); эта возможность оказывается важной при выполнении операций, имеющих вложенную структуру.

В качестве основы для идентификации пользователя GSI использует сертификаты X.509, широко распространённого стандарта для сертификатов инфраструктуры открытых ключей. Для того чтобы приспособить X.509 для поддержки однократной регистрации и делегирования полномочий, GSI определяет прокси-сертификат X.509 [12]. Обычно при выполнении аутентификации GSI использует протокол безопасности транспортного уровня (Transport Layer Security — TLS), являющийся модификацией протокола защищенных сокетов (Secure Socket Level — SSL), хотя и другие протоколы аутентификации на основе технологии открытых ключей могут использоваться для работы с прокси-сертификатами X.509. Протокол удаленного делегирования прокси-сертификатами X.509 надстроен над уровнем TLS.

Рабочая группа по проблемам проектирования Internet (Internet Engineering Task Force — IETF [11, 12, 13]) утвердила предварительный документ, определяющий расширения сертификата X.509 для прокси-сертификата [11]. Всемирный форум Grid (Global Grid Forum — GGF) утвердил предварительные документы, определяющие протокол делегирования для удаленного создания прокси-сертификата X.509 [12] и расширения GSS-API (Generic Security Services API — программный интерфейс унифицированной службы безопасности), позволяющие использовать этот интерфейс для программирования в Grid. Реализации всех алгоритмов защиты в терминах стандарта GSS-API позволяет учесть гетерогенность локальных доменов Grid-системы [8]. Развитая поддержка для ограниченного делегирования была продемонстрирована в прототипах и является существенной составляющей предлагаемого профиля прокси-сертификата X.509. Ограниченное делегирование позволяет одному объекту передать конкретное подмножество «пула» своих привилегий другому объекту. Такое ограничение важно в плане уменьшения вреда при преднамеренном или случайном злоупотреблении делегированным сертификатом.

Защита на уровне сообщений

Защита транспортного уровня реализуется за счет использования самого транспортного механизма (хотя этот метод и обеспечивает внутреннюю защиту сервисов Grid). При интеграции Grid с Web-сервисами (Web-службами), системы Grid переходят к использованию защиты на уровне сообщений. Поскольку последняя подразумевает индивидуальный контроль за каждым сообщением SOAP (Simple Object Access Protocol — простой протокол доступа к объекту) [17], она позволяет применять любые протоколы транспортного уровня; таким образом, можно организовать защиту на разных уровнях в зависимости от важности данных [8].

WS-Security. Компании IBM, Microsoft и VeriSign передали на утверждение в OASIS (Organization for the Advancement of Structured Information Standards) спецификацию защиты Web-сервисов, получившую название WS-Security. Она предлагает платформу передачи сообщений SOAP, служащую для интеграции и поддержки существующих моделей защиты, и набор расширений для SOAP, которые обеспечивают целостность данных и конфиденциальность. Расширение для заголовка сообщений SOAP обеспечивает стандартный, не зависящий от платформы и языка механизм обмена защищенными заверенными сообщениями [8].

Security Assertion Markup Language (SAML). Когда организации совместно используют ресурсы, им необходим общий язык, с помощью которого субъекты Grid могут обмениваться информацией о защите. SAML, утвержденный OASIS в качестве стандарта, определяет язык и протокол для обмена данными об аутентификации и предоставлении прав доступа. Утверждения SAML содержат информацию об аутентификационных ссылках, решения о правах доступа и атрибутах, связанных с указанным субъектом [8].

Правила SAML могут размещаться во внешних хранилищах правил, благодаря чему

виртуальной организации будет проще использовать разнообразные правила, используемые в локальных доменах. SAML определяет интерфейс протокола запросов/ответов, который позволяет клиентам запрашивать утверждения у уполномоченных SAML. Этот протокол, состоящий из форматов сообщений на базе XML, можно легко связать со многими базовыми коммуникациями и транспортными протоколами. Сейчас SAML определяет лишь одну связь — к SOAP через HTTP. Кроме того, временные метки, устанавливаемые для запросов и утверждений SAML, позволяют администраторам Grid связывать временные ограничения с состоянием виртуальной организации и пользовательскими атрибутами. Тем самым отражается динамический характер формирования доверительных отношений в средах Grid [8].

Extensible Access Control Markup Language (XACML). Согласованное представление правил доступа на различных ресурсах является основой реализации защиты. Стандарт Extensible Access Control Markup Language, утвержденный OASIS, определяет базовую схему для выражения правил предоставления прав доступа в формате XML для различных устройств и приложений. Эта схема определяет элементы, требуемые для формулировки правил контроля за доступом, а также предоставляет язык запросов/ответов для передачи запросов и решений. Кроме того, XACML позволяет использовать различные традиционные алгоритмы объединения правил для принятия решений о выборе политики и для объединения правил (возможно, получаемых из разных источников) в единый набор [8].

Подходы к мониторингу поведения пользователей в Grid-системах

Среди существующих работ, посвященных анализу работы пользователей в Grid-системах, следует выделить [18] и [19].

В работе [18] предложен механизм авторизации WAS (Workflow-based Authorization Service), основанный на анализе потока выполнения задач и используемых прав на ресурсах Grid-системы. Идея этого подхода состоит в следующем. При подаче пользователем задачи на выполнение в Grid-системе сервис WAS сначала автоматически анализирует исходный код программ и определяет набор (поток выполнения) прав, которые понадобятся вызываемым функциям программ (наличие возможности анализировать исходный код программы является обязательным условием). Затем этот набор прав наряду с информацией о пользователе отправляется специальному модулю (WAS-server module), который проверяет права пользователя согласно политике безопасности и аутентифицирует корректность сгенерированного набора прав на выполнения задачи. После этого задача и подписанный набор прав отправляется непосредственно на ресурс Grid-системы. Причем ресурс не начнет выполнять задачу без подписанного набора прав. Во время выполнения задачи сервис WAS проверяет права, затребованные задачей, и сравнивает их со сгенерированным набором прав. Если происходит превышение прав, сервис WAS приостанавливает выполнение задачи. Для проверки адекватности предложенного подхода сервис WAS был реализован в программном комплексе Globus Toolkit версий 2.0 и 3.0. Однако при этом не приводятся результаты каких-либо экспериментов по оценке его эффективности. Кроме того, основным недостатком данного сервиса является необходимость наличия исходного кода программ, что в реальных условиях выполняется крайне редко.

В работе [19] предлагается система мониторинга за поведением пользователей в Grid-системе, которая основана на использовании системы MOGAS [16]. Данное средство собирает информацию о состоянии задач, запущенных в программном комплексе Globus Toolkit, и помещает в централизированное хранилище. Однако при этом не обеспечивает информацию о возможных отказах при аутентификации или авторизации. Авторами разработан специальный сценарий для службы Globus gatekeeper (gatekeeper — “привратник”, обеспечивающий безопасное, надёжное создание служб и управление), который собирает информацию об отказах и предоставляет Web-интерфейс для визуализации этой информации. Недостатком данного подхода является то, что не осуществляется анализ данных, собранных во время работы пользователей на ресурсах Grid-системы.

Подходы к оцениванию доступности информации

Важным фактором эффективного выполнения задач Grid-системе является доступность ресурсов и данных. В последние годы лавинообразно возрастает количество атак на доступность ресурсов и информации в компьютерных системах: DoS (denial-of-service) атак [20, 21] и их распределенного варианта — DDoS (distributed denial-of-service). Согласно [22], на сегодняшний день эффективной защиты против подобных атак практически не существует. Основным способом реализации DDoS-атак является использование сетей так называемых компьютеров-зомби (botnets). По данным [23], в начале 2007 г. сети botnets охватывали около 650 миллионов компьютеров. Распределенные атаки, направленные на достижение отказа в обслуживании (DDoS), представляют серьезную угрозу для многих

информационных систем, и гарантированной защиты от них практически не существует. Это приводит к тому, что авторизованные пользователи таких систем не могут своевременно получить доступ к требуемой информации. Подобные факты блокирования работы систем являются причиной огромного ущерба многих организаций во всем мире, в том числе и в Украине.

Одним из наиболее распространенных подходов к оценке доступности ресурсов и информации является общая теория надежности [24, 25], которая развивается с середины 1950-х годов в результате широкого применения методов и средств автоматики и телемеханики. В рамках теории надежности рассматривается понятие *готовности* системы, т.е. состояния работоспособности устройства в произвольно выбранный момент времени. Очевидно, что в программных системах в качестве аналога устройства может рассматриваться некоторый программный компонент или сервис (например, Grid-сервис в Grid-системах). В этом случае можно говорить о готовности или доступности компонента или сервиса и обрабатываемой им информации. Тогда можно считать, что вероятность того, будет ли система и обрабатываемая в ней информация доступной в некоторый момент времени, зависит от средней продолжительности промежутков времени без отказов и восстановлений, которые наблюдаются в данной системе за время ее работы. На основе этого подхода функционирует ПО GridView [26]. GridView представляет собой систему мониторинга и визуализации работы Grid-сервисов. Одна из возможностей данного программного обеспечения — мониторинг доступности Grid-сервисов и целых виртуальных организаций. GridView позволяет собирать информацию о доступности отдельных экземпляров Grid-сервисов, их типов и организации в целом. Информация собирается каждый час и агрегируется по дням, неделям и месяцам [27].

Одной из метрик, используемых в GridView, является состояние экземпляра Grid-сервиса, типа Grid-сервиса и организации в целом. Состояние определяется путем выполнения набора тестов [28]. К одному из недостатков GridView следует отнести недостаточную частоту обновления информации (или выполнения процедуры тестирования (только каждый час)).

Другой подход к оцениванию доступности информации базируется на применении теории нечетких множеств и функций [29]. В рамках этого подхода доступностью считается способность системы ответить на авторизованные запросы в пределах допустимого времени. Однако приемлемое время отклика может зависеть от решаемой прикладной задачи, ее требований и контекста. Основным недостатком этого подхода является невозможность учесть функциональные особенности вычислительной системы и ее компонентов (Grid-сервисов).

Поэтому актуальной является задача создания комплексной методики (информационной технологии) оценивания уровня доступности информации в распределенных Grid-системах, позволяющей учесть функциональные особенности исследуемой системы и проводить подобное оценивание в пределах небольшого промежутка времени. Такая методика предлагается в данной статье. Предлагаемая информационная технология базируется на результатах детального анализа компонентов системы и определении характеристик, которые позволяют объективно оценить уровень доступности информации. Эффективность разработанной методики подтверждается экспериментами, проведенными в процессе исследования Grid-системы для задач исследования Земли в Институте космических исследований НАНУ-НКАУ (ИКИ НАНУ-НКАУ) [30].

Исследование информационной доступности Grid-систем

Информация об уровне загруженности некоторого ресурса рассматриваемой Grid-системы может быть легко получена в процессе его тестирования и анализа отклика соответствующего Grid-сервиса. Для этого, например, можно воспользоваться индексным сервисом подсистемы WS-MDS, входящим в состав ПО GT 4 [31]. Еще одним источником информации о состоянии Grid-системы является подсистема аудита и учета регистрации событий (Audit Logging). Эта система была введена в ПО Globus Toolkit, начиная с версии 4.0.5. Информация, собираемая данной подсистемой, в основном касается подсистемы запуска и выполнения задач GRAM.

Однако пассивный сбор и анализ системных характеристик некоторого информационного ресурса не может гарантировать того, что система работает в нормальном режиме. Для проверки доступности информации предлагается периодически проводить активный эксперимент по проверке работоспособности каждого Grid-сервиса в системе с помощью дополнительного программного компонента, который имитирует работу легитимного пользователя. Такой активный эксперимент позволяет однозначно определить, способен ли каждый конкретный компонент (Grid-сервис) предоставлять и обрабатывать необходимую информацию в пределах допустимого промежутка времени.

При этом необходимо исследовать всю совокупность Grid-сервисов, работающих в

рамках Grid-системы. Для этого были проанализированы и выделены основные типы Grid-сервисов, используемых в Grid-сегменте EOGrid. Также выделены некоторые функциональные особенности каждого типа Grid-сервиса, представляющего интерес с точки зрения вклада, который вносит каждая конкретная функция в обработку информации, за которую отвечает этот тип сервисов.

На основе функциональных особенностей Grid-сервисов каждого типа можно предложить сценарии тестирования каждой из описанных функций \. Такие планы тестирования составляют основу активного эксперимента по проверке доступности всех Grid-сервисов Grid-системы.

Для реализации в соответствии с методологией [32] активного эксперимента проверки доступности Grid-сервисов системы EOGrid, которая функционирует в ИКИ НАНУ-НКАУ, были сформулированы и описаны соответствующие прецеденты. При этом на начальном этапе были выделены основные требования, общие для всех прецедентов (таблица 1).

Таблица 1. Основные требования реализации прецедентов

Рамки	Программная инфраструктура EOGrid
Уровень	Системный
Основной исполнитель.	Подсистема обеспечения безопасности и аудита
Заинтересованные лица и их требования	– *Модератор виртуальной организации.* Хочет постоянно обладать полной информацией о функционировании Grid-сегмента. – *Системный администратор Grid-ресурса.* Хочет своевременно получать информацию о качестве функционирования своих аппаратных и программных средств. – *Пользователь Grid-ресурса.* Хочет работать с системой, которая функционирует без сбоев и всегда доступна.
Технологии	При реализации данных прецедентов должны использоваться технологии и средства разработки, поддерживающие Web- или WSRF-сервисы, механизм обмена сертификатами, документами XML, реализацию шаблона MainFacade, абстрактного интерфейса Factory и объектно-реляционное преобразование.
Предусловия	Для выполнения задачи подсистемой обеспечения безопасности и аудита успешно получен временный цифровой сертификат
Постусловия	При успешном завершении данного прецедента гарантируется доступность тестируемого Grid-сервиса

Набор прецедентов включает:

– Прецедент П1. Проверка доступности Grid-сервиса WS-MDS;

– Прецедент П2. Проверка доступности Grid-сервисов Grid-FTP и RFT

– Прецедент П3. Проверка доступности Grid-сервиса GRAM

Для реализации предложенной выше методики проверки доступности сервисов было разработано прикладное программное обеспечение для проведения периодического тестирования ключевых Grid-сервисов и ресурсов Grid-системы для задач исследования Земли ИКИ НАНУ-НКАУ.

В качестве хранилища для полученных результатов тестирования, измерений загруженности ресурсов и оценок доступности информации в данной вычислительной системе использовалась реляционная СУБД PostgreSQL.

Данная реализация программной системы сбора и обработки данных мониторинга доступности Grid-сервисов построена на основе каркаса Spring [33]. Spring представляет собой систему управления компонентами на базе конфигурационных файлов и атрибутов (метаданных, которые сохраняются непосредственно в коде программы).

Каркас Spring был интегрирован с системой объектно-реляционного отображения Hibernate [34] при построении так называемых объектов доступа к данным (Data Access Object — DAO) и управлении транзакциями. Выбор такой архитектуры позволил существенно сократить время, необходимое на разработку подсистемы доступа к данным, благодаря исключению детальных SQL-запросов к базе данных.

Центром сбора информации в данной программной системе является объект класса TaskRunner (рис. 1). Он отвечает за периодическое получение следующего набора задач (тестов) от сконфигурированной фабрики (объект класса, который реализует интерфейс TaskFactory) и поддержку очереди задач.

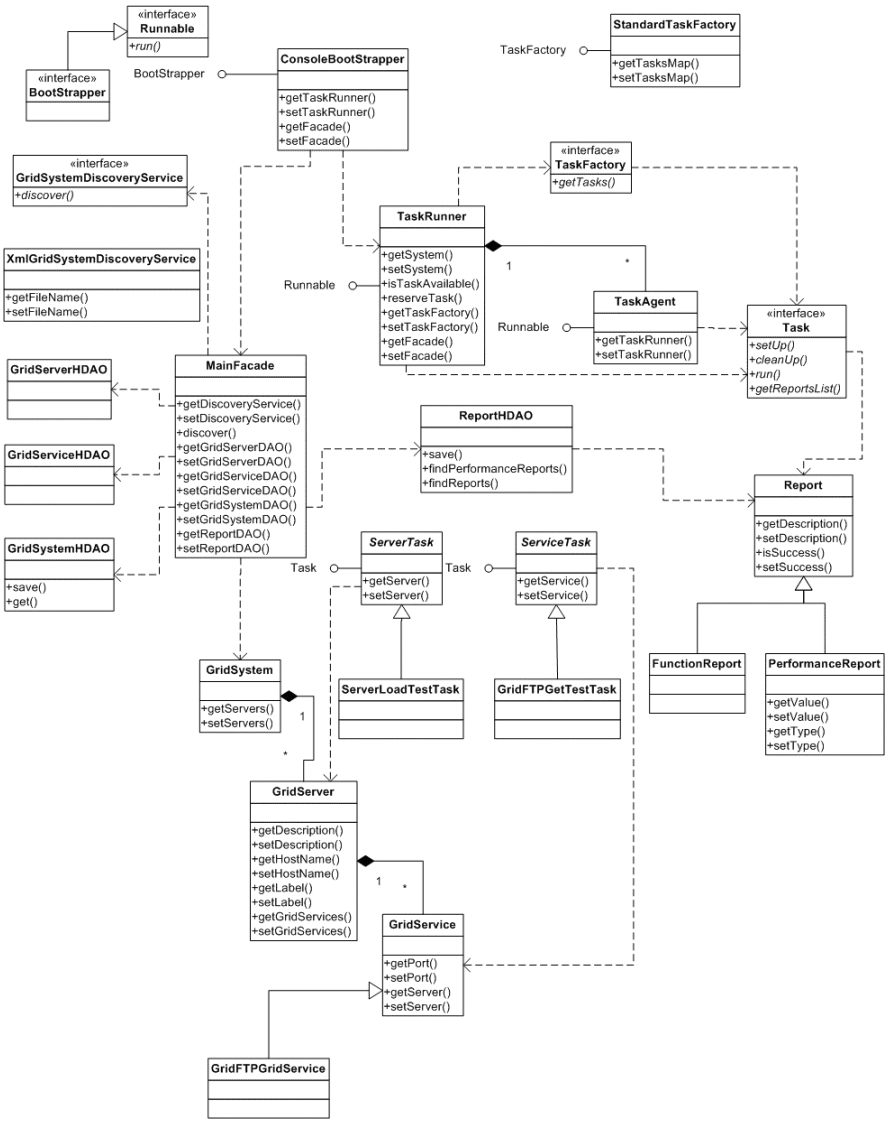

Рисунок 1 - Диаграмма классов программного модуля мониторинга и анализа доступности Grid-сервисов

Тестовые задачи представлены объектами классов, которые отвечают за реализацию проверки той или другой функции некоторого типа Grid-сервиса и наследуются от абстрактного класса ServiceTask. Задачи по сбору информации об уровне загруженности процессора и объеме использованной памяти некоторого ресурса представлены объектами классов, которые наследуются от абстрактного класса ServerTask.

Объекты класса TaskAgent выполняются в своем потоке и получают каждую следующую задачу из очереди задач, после чего ее выполняют. Результаты работы задач и выполнения тестов сохраняются в базе данных с использованием объектов доступа к данным (DAO), которые доступны посредством свойства объекта класса MainFacade. Класс MainFacade представляет собой главный интерфейс бизнес-логики данной программной системы.

При запуске данной программной системы все ссылки на интерфейсы и абстрактные классы получают экземпляры конкретных классов благодаря системе Spring. Далее происходит процесс загрузки системы, который контролируется объектом класса, реализующий интерфейс BootStrapper. Таким образом, происходит инициализация компонентов системы.

Поскольку для работы в данной системе необходим цифровой сертификат пользователя, перед началом работы каждого теста или задачи необходимо получить временный сертификат. Это происходит с использованием сервиса MyProxy. Кроме того, необходимо уничтожить полученный временный сертификат после окончания работы задачи или теста.

Разработанное ПО мониторинга и оценивания доступности информации в Grid-системе позволяет эффективно проводить периодический активный и пассивный аудит работы всех компонентов системы.

На основе полученной информации можно ответить на вопрос, способна ли система выполнять свою основную задачу, т.е. является ли доступной предоставляемая ею информация.

Статистический подход к анализу поведения пользователей в Grid-системах

При построении моделей поведения пользователей компьютерных систем (не только Grid-систем) можно выделить следующие общие этапы: (1). Сбор и предварительная обработка данных о работе пользователей. (2). Анализ данных для выделения информативных признаков или уменьшения размерности данных (создание так называемого профиля пользователя). (3). Построение модели. (4). Верификация модели и интерпретация полученных результатов.

Предлагаемая модель поведения пользователя Grid-системы строится на основе нейросетевого подхода [35]. При этом рассматриваются все перечисленные выше этапы построения модели. После того, как для каждого пользователя Grid-системы построена и верифицирована модель, выявление аномальной деятельности должно происходить следующим образом: если результаты текущей работы пользователя соответствуют ранее построенной эталонной модели, то такое поведение можно считать нормальным. В противном случае — аномальным. Блок-схема процесса выявления аномалий в деятельности пользователей Grid-систем представлена на рис. 2.

Более строго задачу выявления аномалий в поведении пользователей Grid-систем можно сформулировать следующим образом.

Пусть для каждого пользователя Grid-системы $u \in U$ (U — множество пользователей) имеется набор статистических данных и информативных признаков $x \in X$ о запуске задач в Grid-системе. Для каждого пользователя Grid-системы необходимо построить модель, осуществляющую преобразование

$$F: X \to \{0, 1\}$$

т.е., классифицирующую поведение пользователя на нормальное и аномальное на основе доступной инфомации.

Для анализа статистических данных о работе пользователя и выявления аномалий предлагается использовать нейронные сети [36]. Применение нейронных сетей обеспечивает интеллектуальный и робастный подход к анализу и обобщению данных о работе пользователя.

Среди многочисленных нейросетевых парадигм [36] для решения поставленной задачи наилучшим выбором является многослойная сеть прямого распространения информации, представляющая собой иерархическую структуру взаимосвязанных простых обрабатывающих элементов (исполнительных нейронов). Связи между элементами одного слоя и обратные связи в сети отсутствуют. Использование сетей данного вида обусловлено тем, что, согласно теореме Колмогорова, они являются универсальными апроксиматорами [37–39] и могут эффективно применяться как для решения задач прогнозирования, так и классификации.

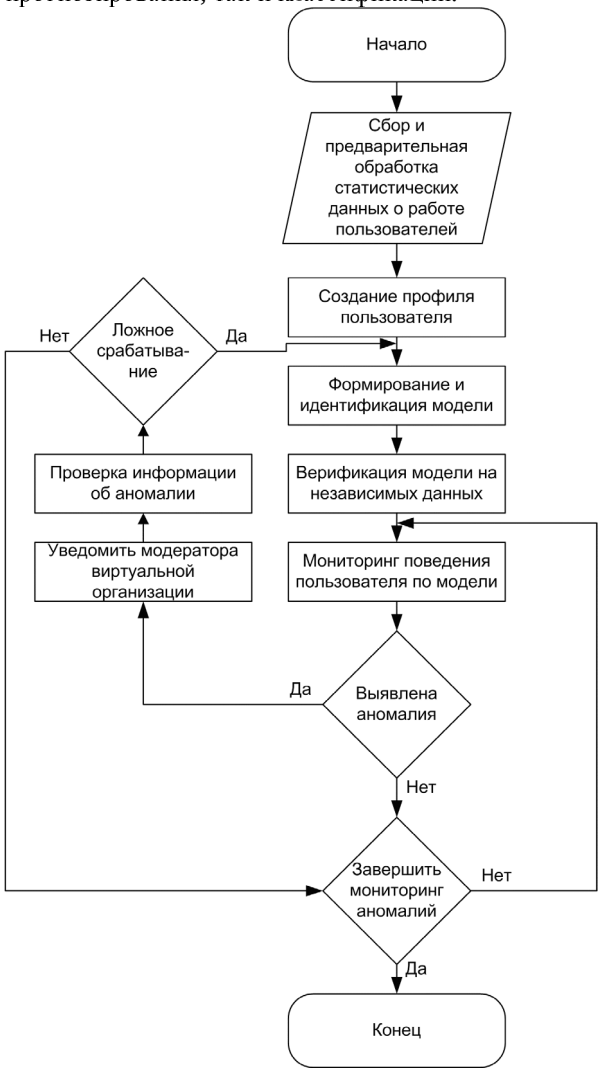

Рисунок 2 - Блок-схема процесса выявления аномалий в деятельности пользователей Grid-систем

Таким образом, задача структурной и параметрической идентификации модели поведения пользователя Grid-систем сводится построению нейронной сети прямого распространения и ее обучению таким образом, чтобы на основе доступной информации отнести поведение пользователя к одному из классов: нормальному или аномальному. При этом ожидаемый выход нейронной сети может принимать два значения: 1 — для нормального поведения пользователя и 0 — для аномального, т.е. нейронная сеть должна функционировать в качестве классификатора.

При работе пользователей в Grid-среде существуют свои особенности, связанные с тем, что пользователь выполняет небольшое количество трудоемких задач на высокопроизводительных ресурсах Grid-системы. Поэтому целесообразно собирать и анализировать информацию о запускаемой задаче (т.е. аналогом

понятия сеанса, используемого в [40, 41], в предлагаемой модели является процесс запуска пользователем задач в Grid-среде). При построении профиля и модели поведения пользователя будем учитывать всю доступную измерению информацию. На вход нейросетевой модели поступает следующая информация:

– Ресурс сегмента, виртуальную организацию и брокер ресурсов будем нумеровать целыми числами 1, 2, 3, ...;

– время работы процессора, полное время выполнение задачи и разница между временем начала выполнения задачи на ресурсе Grid-системы и временем отправки задачи в Grid-систему измеряется в секундах;

– отношение времени работы процессора к общему времени выполнения задачи представляется числом из отрезка [0; 1];

– статус завершения задачи описывается бинарным значением: 0 — успешное выполнение и 1 — с ошибкой;

– время отправки (создания) задачи в Grid-систему измеряется в минутах (с начала дня) и нормируется на 24 часа;

– оперативная и виртуальная память измеряются в Гбайтах.

В работе [36] показано, что если при обучении нейронной сети желаемый выход принимает два значения (например, 0 и 1; т.е. нейронная сеть разделяет входное пространство на два класса), то при подаче на ее вход независимого образа на выходе будет получена вероятность принадлежности этого образа к одному или другому классу. Таким образом, значение $\Delta_{s_t^u}$ (выход нейронной сети) будет принадлежать отрезку [0; 1] и определять вероятность нормального (соответствующего модели) поведения пользователя.

Предложенная модель поведения пользователя имеет следующие преимущества:

– независимость от количества пользователей в системе, поскольку для каждого пользователя строится своя нейросетевая модель;

– адаптация к изменению поведения пользователей;

– использование интеллектуальных методов обработки данных.

Для проверки эффективности предложенной модели пользователей Grid-системы была проведена серия экспериментов. При этом использовались данные, полученные в результате работы пользователей в обучающей системе GILDA (https://gilda.ct.infn.it/) европейского проекта EGEE (http://www.eu-egee.org/). Для того чтобы проверить, насколько нейронная сеть была способна отличить поведение одного пользователя от другого, использовалась процедура подмены пользователя. На вход нейронной сети, которая была обучена для одного пользователя (легального), подавались данные другого пользователя (нелегального). Так имитировалась ситуация, когда нелегальный пользователь работает под именем (учетной записью) легального пользователя. Результаты экспериментов показали, что процент правильной классификации для легального пользователя на тестовой выборке составил 99,14% (соответствует ошибке первого рода). В случае моделирования процедуры подмены пользователя процент правильной классификации нелегального пользователя составил 99,30% (соответствует ошибке второго рода). Приведенные результаты показывают, что предложенная модель позволяет уверенно обнаружить подмену пользователя, поэтому является достаточно эффективной.

Выводы

В данной статье проанализированы средства обеспечения безопасности Grid-систем, и исследованы два аспекта безопасности функционирования Grid-систем: аномалий в деятельности пользователей и мониторинга доступности ресурсов и данных в Grid-системе.

Анализ существующих методов анализа и моделей поведения пользователей показал, что в большинстве своем они разработаны для компьютерных сетей. С учетом особенностей Grid-систем и требований к безопасности при выполнении распределенных вычислений важнейшей задачей является адаптация этих методов и разработка новых для Grid-систем.

Предложен новый подход к выявлению аномалий в поведении пользователей Grid-систем, который основан на нейросетевом анализе статистических данных о задачах, выполняемых пользователем в Grid-системе. Предложенная модель верифицирована на данных, которые были собраны системой GridICE при работе реальных пользователей в Grid-системе GILDA-EGEE и показала свою эффективность по выявлению аномальной деятельности пользователей (более 99% правильной классификации).

Предложена информационная технология проверки доступности информации Grid-системы на основе активного эксперимента, обеспечивающая мониторинг доступности данных и сервисов в реальном режиме времени. При этом для выполнения необходимой оценки используются как данные системных журналов регистрации (пассивного эксперимента), так и результаты активных сценариев тестирования компонентов Grid-системы. В отличие от существующих систем мониторинга ресурсов предлагаемое средство мониторингна позволяет учесть специфику конкретной Grid-системы и проверить доступность каждого сервиса. Разработанная комплексная методика реализована в виде прикладного ПО мониторинга доступности Grid-сервисов и ресурсов.

Работа выполнена при поддержке гранта УНТЦ-НАНУ «Разработка Grid-технологий интеграции данных разной природы» (№4928).

Литература

1. Putting Earth-Observation on the Grid / L.Fusco, P. Goncalves, J. Linford, M. Fulcoli, A. Terracina, G. D'Acunzo // ESA Bulletin. — 2003. — **114**. — P. 86–91.
2. Fusco L. Open Grid Services for Envisat and Earth Observation Applications / R. Cossu, C. Retscher // In: Plaza AJ, Chang C-I (ed) High performance computing in remote sensing, 1st edn. — Taylor & Francis Group, New York, 2007. — P. 237-280.
3. Shelestov A.Yu. Grid Technologies in Monitoring Systems Based on Satellite Data / A.Yu. Shelestov, N.N. Kussul, S.V. Skakun // J. of Automation and Information Science. — 2006. — **38**, N 3. — P. 69-80.
4. CMS Requirements for the Grid : Proc. of the Int. Conf. on Computing in High Energy and Nuclear Physics (CHEP2001) / K. Holtman — citeseer.ist.psu.edu/article/holtman01cms.html.
5. Peltier S.T. The telescience portal for advanced tomography applications / S.T. Peltier — J. of Parallel and Distributed Computing: Computational Grid. — 2002. — **63**, N 5. — P. 539–550.
6. Annis J. Applying chimera virtual data concepts to cluster finding in the sloan sky survey / J. Annis, Y. Zhao — ftp://info.mcs.anl.gov/pub/tech_reports/reports/P978.pdf.
7. Foster I. The Anatomy of the Grid. Enabling Scalable Virtual Organizations / I. Foster, C. Kesselman, S. Tuecke // Intern. J. Supercomputer Applications. — 2001. — **15**, N 3. — P. 200–222.
8. Рамакришнан Л. Защита Grid / Л. Рамакришнан // Открытые системы. — 2004. — № 6. — С. 63-68.
9. Cornwall L.A. Authentication and authorization mechanisms for multi-domain grid environments / L.A. Cornwall, J. Jensen, D.P. Kelsey // J. of Grid Computing. — 2004. — **9**. — P. 301–311.
10. A security architecture for computational grids: Proc. of ACM Conf. on Computers and Security. / I. Foster, C. Kesselman, G. Tsudik, S. Tuecke — 1998. — P. 83–91.
11. Adams C. Understanding PKI: concepts, standards, and deployment considerations. / C. Adams, S. Lloyd — London: Addison-Wesley, 2002. — 352 p.
12. IETF – Public-Key Infrastructure (X.509) (pkix), 2005. — www.tools.ietf.org/wg/pkix.
13. IETF – Transport Layer Security (tls), 2005. — www.tools.ietf.org.
14. Tulloch M. Microsoft Encyclopedia of Security/ M. Tulloch— Redmond, Washington: Microsoft Press, 2003. — 414 p.
15. GridICE. — http://gridice.forge.cnaf.infn.it.
16. MOGAS. — http://ntu-cg.ntu.edu.sg/pragma/index.jsp.
17. Simple Object Access Protocol (SOAP) 1.1. W3C, Note 8, 2000.
18. Seung-Hyun K., Kyong H.K., Jong K., Sung-Je H., Sangwan K. Workflow-Based Authorization Service in the Grid. J. of Grid Computing, 2004, Num. 2, P. 43–55.
19. Shingo T., Susumu D., Shinji S. A user-oriented secure filesystem on the Grid // The 3rd IEEE/ACM Int. Symp. on Cluster Computing and the Grid (CCGrid 2003), May, 2003.
20. Denial-of-service attack. — http://en.wikipedia.org/wiki/Denial-of-service_attack.
21. Уланов А. Защита от DDoS-атак: механизмы предупреждения, обнаружения, отслеживания источника и противодействия / А. Уланов, И. Котенко // Информационно-методический журнал "Защита информации. Инсайд". — 2007. — № 1. — С. 20–30.
22. Handley M. Internet. Denial-of-Service Considerations. RFC4732. / M. Handley, E. Rescorla — IETF. Network Working Group, 2006. — 38 p.
23. Markoff J. Attack of the Zombie Computers Is Growing Threat / J. Markoff // The New York Times. — 2007. — **1**. — P. 10–15.
24. Синопальников В.А. Надежность и диагностика технологических систем. / В.А. Синопальников, С.Н. Григорьев — М:. "Высшая школа", 2005. — 343 с.
25. Матвеевский В.Р. Надежность технических систем. [Учебное пособие.] / В.Р. Матвеевский — М.: МГИЭМ, 2002. — 113 с.
26. Kalmady R. GridView: a Grid monitoring and vizualization tool. / R. Kalmady, D. Sonvane, K. Bhatt — https://twiki.cern.ch/twiki/pub/LCG/GridView/.
27. Sonvane D. Computation of Service Availability Metrics in Gridview. / D. Sonvane, R. Kalmady, P. Chand. — https://twiki.cern.ch/twiki/pub/LCG/GridView/.
28. Leslie M. SAM Technical Assessment of Sam Test Harness. / M. Leslie, A. Lyon, S. Stonjek — http://home.fnal.gov/~stdenis/sam/assessment/.
29. Tryfonas T. An Alternative Model for Information Availability./ T. Tryfonas — http://www.unob.cz/spi/2007/presentace/2007-May-03/03-Tryfonas-Alternative_Model.ppt.
30. Куссуль Н.Н., Шелестов А.Ю. Grid-системы для задач исследования Земли. Архитектура, модели и технологи. — К.: "Наукова думка", 2008. - 452 с.
31. Schopf J. M. Monitoring the grid with the Globus Toolkit MDS4 / J. M. Schopf, L. Pearlman, N. Miller // Journal of Physics. — 2006. — **46**. — P. 521–525.
32. Ларман К. Применение UML 2.0 и шаблонов проектирования. / К. Ларман — М.: Вильямс, 2006. — 736 с.
33. The Spring Framework - Reference Documentation. / R. Johnson, J. Hoeller, A. Arendsen, C. Sampaleanu. —

http://springframework.org.
34. Bernard E. Hibernate — Relational Persistence for Idiomatic Java. Reference Documentation. / E. Bernard, A. Patricio, J.C. Rousseau — http://www.hibernate.org.
35. Shelestov A. Intelligent Model of User Behavior in Distributed Systems / A. Shelestov, S. Skakun, O. Kussul // International Journal on Information Theory and Applications. — 2008. — **15**(1) . — P. 70-76
36. Хайкин С. Нейронные сети: полный курс/ С. Хайкин;[Пер. с англ.] — [2-е издание] — Издательский дом «Вильямс», 2006. — 1104 с.
37. О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных : Докл. АН СССР / А.Н. Колмогоров — 1956. — Т. 108, No. — С. 179–182.
38. Арнольд В.И. О функциях трех переменных : Докл. АН СССР. / В.И. Арнольд — 1957. — Т. 114, No. 4. — С. 679–681.
39. О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного: Докл. АН СССР // А.Н. Колмогоров — 1957 — Т. 114. — No. 5. — С. 953-956.
40. Скакун С.В. Нейросетевая модель пользователей компьютерных систем / С.В. Скакун, Н.Н. Куссуль // Кибернетика и вычисл. техника. — 2004. — Вып. 143. — С. 55–68.
41. Непараметрическая идентификация комплексной нейросетевой модели поведения пользователей компьютерных систем / Скакун С.В. // Кибернетика и вычисл. техника. — 2005. — Вып. 147. — С. 45–69.

Поступила в редколлегию 10.03.2009