УДК 681.63, 519.872

А.Ю. Шелестов, С.И. Лавренюк

ОБЕСПЕЧЕНИЕ БЕЗОПАСНОСТИ GRID-СИСТЕМ НА ОСНОВЕ МОДЕЛИ ПОВЕДЕНИЯ ПОЛЬЗОВАТЕЛЕЙ

Приведен обзор моделей и методов обеспечения безопасности в Grid-системах, базирующихся на мониторинге поведения пользователей. Предложен подход к обеспечению безопасности Grid-систем на основе построения профиля пользователя с параметрами запускаемых им задач. Приведены результаты экспериментов на основе реальных данных, полученных в Grid-системе GILDA-EGEE.

Введение

В настоящее время решение сложных вычислительных задач в области наблюдения Земли из космоса [1–3], физики высоких энергий [4], биоинформатики [5], астрономии [6] и многих других сферах научной и прикладной деятельности невозможно без использования сложных распределенных систем, в частности Grid-систем. Технология Grid предполагает использование программного обеспечения среднего уровня (middleware), предназначенного для объединения распределенных информационных и вычислительных ресурсов различных административных доменов в рамках единой виртуальной организации (ВО) [7]. При этом Grid-платформа призвана решить такие задачи, как гибкое, безопасное и согласованное совместное использование ресурсов, а также обеспечение параллельных высокопроизводительных вычислений и распределенной обработки данных. Очевидно, что одной из важнейших задач при разработке подобных сложных систем является реализация механизмов обеспечения безопасности [7], в частности, аутентификации и авторизации, обмена сертификатами, обеспечения конфиденциальности и целостности данных, а также аудит и мониторинг ресурсов и пользователей [7–9]. 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 (Public Key Infrastructure — PKI) [11, 12]. GSI-инфраструктура поддерживает одноразовую регистрацию (single sign on), делегирование полномочий и обмен сертификатами.

Вместе с тем следует отметить, что одним из наиболее важных аспектов обеспечения безопасности является мониторинг действий пользователей при работе с удаленными ресурсами Grid-системы.

Анализ моделей безопасности Grid-систем

В настоящее время существует достаточно много средств мониторинга состояния ресурсов Grid-системы и запускаемых задач (например, GridICE [13] и MOGAS [14]). 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.

Рассмотрим существующие средства обеспечения безопасности в Grid-системах подробнее. В настоящее время при разработке большинства Grid-систем используется программное обеспечение Globus Toolkit. Как уже упоминалось выше, для обеспечения безопасности в данном программном обеспечении (следовательно, и в прикладных Grid-системах) используется инфраструктура GSI [10]. Основанный на технологии открытых ключей протокол GSI обеспечивает аутентификацию и однократную регистрацию пользователей, а также ограниченный набор средств делегирования полномочий. Для идентификации пользователей в инфраструктуре GSI используются сертификаты X.509 [15] рабочей группы IETF (Internet Engineering Task Force [11]). Специалистами этой же группы определен также документ, регламентирующий расширение сертификатов X.509 для обеспечения поддержки прокси-сертификатов [15].

При реализации в Grid-среде Web-сервисов (или Web-служб) защита обеспечивается уже на уровне сообщений протокола SOAP (Simple Object Access Protocol — простой протокол доступа к объектам) [16].

Таким образом, системные средства защиты в Grid-системах развиты достаточно хорошо. Отдельным вопросом обеспечения безопасности Grid-систем является анализ поведения пользователей [17, 18].

В работе [17] предложен механизм авторизации WAS (Workflow-based Authorization Service), основанный на анализе потока выполнения задач на ресурсах Grid-системы и используемых при этом пользовательских привилегий (или разрешений). Основная идея этого подхода состоит в следующем. При отправке пользователем задачи на выполнение в Grid-систему сервис WAS сначала автоматически анализирует исходный код программы и определяет набор разрешений, которые понадобятся для ее выполнения. (Обязательное условие — возможность анализа исходного кода программы.) Затем полученный набор разрешений наряду с информацией о пользователе передается специальному модулю (WAS-server module), который проверяет набор пользовательских разрешений на соответствие принятой политике безопасности и оценивает их корректность. После успешного прохождения проверки задача и набор разрешений отправляются непосредственно на ресурс Grid-системы. В процессе выполнения задачи сервис WAS осуществляет мониторинг запрашиваемых ею разрешений и сравнивает их с набором, сгенерированным ранее. При обнаружении несоответствия выполнение задачи сервисом WAS прерывается.

Для проверки адекватности предложенного подхода сервис WAS был реализован в программном комплексе Globus Toolkit версий 3.*x* и 4.*x*. Однако результаты каких-либо проведенных экспериментов по оценке его эффективности в литературе не приводятся.

В [18] предложен модуль мониторинга поведения пользователей Grid-системы, основанный на применении механизма MOGAS [14]. Он позволяет собирать данные о состоянии задач, запущенных в Grid-среде на основе Globus Toolkit, и размещать ее в централизованном хранилище. Однако информация о возможных отказах при аутентификации или авторизации при этом не предоставляется. Авторами предложен сценарий для службы Globus

gatekeeper, который позволяет собирать информацию об отказах и просматривать ее через Web-интерфейс. Существенный недостаток данного подхода — отсутствие средств анализа собранных данных, что весьма снижает ценность предложенного подхода.

Идея предлагаемого подхода

Анализ существующих средств обеспечения безопасности в Grid-системах свидетельствует о том, что методы и средства мониторинга поведения пользователей в Grid-системах в настоящее время развиты недостаточно, а имеющиеся средства мониторинга состояния ресурсов Grid-систем (например, GridICE, MOGAS) не обеспечивают необходимой функциональности. Лишь в некоторых работах (например, в [17, 18]) рассматриваются вопросы, связанные с мониторингом деятельности пользователей на основе анализа потока выполнения задач, требуемых для этого разрешений, а также об отказах системы.

В настоящее время разработано достаточно много методов анализа и моделей поведения пользователей компьютерных сетей. Так, в [19, 20] предложена комплексная модель поведения пользователей компьютерных систем, которая состоит из трех компонентов (интерактивной (прогнозной) составляющей, сеансовой (статистической) составляющей и модуля анализа трендов) и позволяет учесть как динамические, так и статистические свойства поведения пользователей, а также возможные тренды его поведения. Вместе с тем работа пользователей в Grid-системе имеет свою специфику и особенности (решение сложных задач на распределенных высокопроизводительных ресурсах). Поэтому в данной работе ставится задача модифицировать комплексную модель и учесть специфику Grid-вычислений.

При построении моделей поведения пользователей 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) разработка методов обработки данных и построение модели;

4) верификация модели и интерпретация полученных результатов.

В данной статье модель поведения пользователя Grid-системы строится на основе нейросетевого подхода. При этом рассматриваются все перечисленные выше этапы построения модели. Такая модель должна обеспечить возможность выявления характерных (аномальных) действий пользователя Grid-системы. Если результаты работы пользователя соответствуют ранее построенной модели, то такое поведение можно считать нормальным, в противном случае — аномальным.

Структура модели

Для анализа статистических данных о работе пользователя для выявления аномалий предлагается использовать нейронные сети [21]. Применение нейронных сетей обеспечивает интеллектуальный и робастный подход к

анализу и обобщению данных о работе пользователя. В общем случае существуют различные нейросетевые парадигмы. Для решения поставленной задачи наилучшим выбором является многослойная сеть прямого распространения. Это обусловлено тем, что согласно теореме Колмогорова они являются универсальными аппроксиматорами [21] и могут эффективно применяться как для решения задач прогнозирования, так и классификации. Аспекты построения нейронных сетей прямого распространения и методы настройки ее коэффициентов подробно описаны в [21].

В работах [19, 20] сеансовая модель основывалась на анализе следующей информации о работе пользователя: количество команд, выполненных пользователем в течение сеанса; результат интерактивной модели (относительное количество правильно спрогнозированных команд за сеанс), номер компьютера в сети, за которым работал пользователь; продолжительность сеанса; время начала сеанса. При этом за сеанс пользователь может выполнять десятки (а иногда и сотни) команд, необходимых для решения его задач. 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 сеанса в исходной модели будет процесс запуска задачи в предлагаемой сеансовой модели). Таким образом, при построении модели поведения пользователя предлагается учитывать следующую информацию:

$$\{S, ET, CPU, WT, CW, ES, CT, STD, RAM, VM, VO, RB\}, \quad (1)$$

где S (Site) — сайт, на котором выполнялась задача; ET (Execution Target) — ресурс сайта, на котором выполнялась задача; CPU (CPU Time) — время работы процессора ресурса при выполнении задачи; WT (Wall Time) — полное время выполнения задачи; CW (CPUWall = CPU/W) — отношение времени работы процессора к общему времени выполнения задачи; ES (ExitStatus) — статус завершения задачи (успешное выполнение или с ошибкой); CT (Creation Time) — время отправки (создания) задачи в Grid-систему; STD (Start Time Difference) — разница между временем начала выполнения задачи на выбранном ресурсе Grid-системы (выбор конкретного ресурса, на котором будет выполняться задача, обеспечивается брокером ресурсов) и временем отправки задачи в Grid-систему; RAM (RAM Used) — используемая оперативная память; VM (Virtual Memory Used) — используемая виртуальная память; VO (Virtual Organization Name) — принадлежность к виртуальной организации; RB (Resource Broker Hostname) — брокер ресурсов, используемый для распределения задачи.

Этот набор данных является входным признаком для выявления нормальной или аномальной работы пользователя. Для решения этой задачи предлагается применять нейронную сеть прямого распространения, т.е. для каждого пользователя Grid-системы необходимо построить нейронную сеть, которая обучается 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. Другими словами, нейронная сеть должна функционировать в качестве классификатора.

Пусть $u \in \mathrm{U}$ — некоторый пользователь Grid-системы (U — множество пользователей), s_t^u ($t \in \{1, 2, \ldots\}$) — набор задач, запущенных пользователем u в Grid-системе, для которых имеется следующий набор данных (1):

$$\mathbf{x}_{s_t^u} = \{\mathrm{S}_{s_t^u}, \mathrm{ET}_{s_t^u}, \mathrm{CPU}_{s_t^u}, \mathrm{WT}_{s_t^u}, \mathrm{CW}_{s_t^u}, \mathrm{ES}_{s_t^u}, \mathrm{CT}_{s_t^u}, \mathrm{STD}_{s_t^u}, \mathrm{RAM}_{s_t^u}, \mathrm{VM}_{s_t^u}, \mathrm{VO}_{s_t^u}, \mathrm{RB}_{s_t^u}\}.$$

Тогда выход нейронной сети по завершении задачи s_t^u определяется следующим соотношением (рис. 1):

$$\Delta_{s_t^u} = F(\mathbf{x}_{s_t^u}),$$

где F — нелинейное преобразование нейронной сети прямого распространения; $\mathbf{x}_{s_t^u}, \Delta_{s_t^u}$ — вход и выход сети соответственно (в данном случае размерность вектора $\mathbf{x}_{s_t^u}$ составляет 12, а $\Delta_{s_t^u}$ — 1).

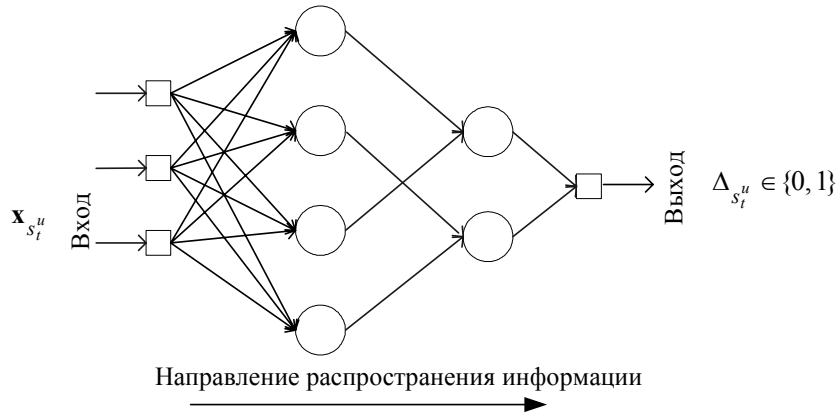

Рис. 1. Структура сеансовой модели

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 выполнения 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 целесообразно представлять 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 гигабайтах.

В работе [21] 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, итак, значение $\Delta_{s_i^u}$ будет принадлежать отрезку [0; 1] и определять вероятность нормального (соответствующего модели) поведения пользователя.

Следует выделить такие преимущества модели пользователя:

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

• адаптация к изменению поведения пользователей;

• использование интеллектуальных методов обработки данных.

Поскольку нейросетевые модели по своей природе индукционные, первоочередную роль в их построении играют экспериментальные данные. Рассмотрим структуру данных и источники их получения подробнее.

Описание структуры данных

При построении модели использовались данные, полученные в результате работы пользователей в обучающей системе GILDA (https://gilda.ct.infn.it/) европейского проекта EGEE (http://www.eu-egee.org/). Система GILDA объединяет ресурсы 12 организаций, насчитывая в общей сложности 112 процессоров и возможность хранения до 5,2 Тбайт данных.

Для мониторинга состояния ресурсов и задач в Grid-системе GILDA используется распределенная система GridICE. Она интегрируется с локальной системой мониторинга ресурса и предоставляет стандартный интерфейс для отображения данных на уровне Grid-системы. Распространение данных может выполняться на двух уровнях иерархии: локальном (конкретного ресурса) и всей Grid-системы. При этом существует несколько подходов для отображения данных мониторинга: в графическом или текстовом виде посредством Web-интерфейса или с использованием формата XML. Среди полезных свойств системы GridICE можно выделить следующие:

— автоматическое обнаружение новых ресурсов, мониторинг которых необходимо проводить посредством использования службы Grid Information Service;

— мощный инструментарий для отображения данных мониторинга через Web-интерфейс;

— наличие служб уведомления;

— полный набор метрик мониторинга (для отдельного ресурса и всей системы в целом);

— поддержка таких систем управления задачами, как OpenPBS, Torque, LSF;

— предоставление данных в формате XML;

— открытый код.

В таблице приведены данные о задачах, которые предоставляет система мониторинга GridICE.

Таблица

Название	Описание
Job LocalID	Локальный идентификатор задачи
Name	Название задачи
JobStatus	Статус выполнения задачи (E — выполнена, W — ожидание, R — в процессе выполнения)
LocalOwner	Владелец задачи
Execution Target	Ресурс сайта, на котором выполнялась задача
CPU Time	Время работы процессора ресурса при выполнении задачи
Wall Time	Полное время выполнения задачи
CPUWall	Отношение времени работы процессора к общему времени выполнения задачи
Exit Status	Статус завершения задачи (успешное выполнение или с ошибкой)
Creation Time	Время отправки (создания) задачи в Grid-систему
Start Time	Время начала выполнения задачи
Start Time Difference	Разница между временем начала выполнения задачи на выбранном ресурсе Grid-системы (выбор конкретного ресурса обеспечивается брокером ресурсов) и временем отправки задачи в Grid-систему
End Time	Время завершения задачи
RAM Used	Используемая оперативная память
Virtual Memory Used	Используемая виртуальная память
Virtual Organization Name	Принадлежность к виртуальной организации
Site	Сайт организации, на котором выполнялась задача
Resource Broker Hostname	Брокер ресурсов, который использовался для распределения задачи
GlobalID	Глобальный идентификатор задачи

Ниже приведен пример данных в формате XML с описанием задачи:

```
<Job LocalID="5794">
<Name>STDIN</Name>
<JobStatus>E</JobStatus>
<LocalOwner>gilda001</LocalOwner>
<ExecutionTarget>iceage-wn-13</ExecutionTarget>
<CPUTime UnixTime="7">00:00:07</CPUTime>
<WallTime UnixTime="27">00:00:27</WallTime>
<CPUWall>0.25925925925926</CPUWall>
```

```
    <ExitStatus>0</ExitStatus>
    <CreationTime                    UnixTime="1175270319">2007-03-30
17:58</CreationTime>
    <StartTime UnixTime="1175270320">2007-03-30 17:58</StartTime>
    <StartTimeDiff UnixTime="1">00:00:01</StartTimeDiff>
    <EndTime UnixTime="1175270347">2007-03-30 17:59</EndTime>
    <RAMUsed>19576</RAMUsed>
    <VirtualUsed>45000</VirtualUsed>
    <VOName>gilda</VOName>
    <Site>ICEAGE-CATANIA</Site>
    <GlobalID>https://glite-
rb.ct.infn.it:9000/Ba5Xi27XOjie4e2sk8ZJug</GlobalID>
    <RBHostname>glite-rb.ct.infn.it</RBHostname>
    </Job>
```

Для проведения экспериментов и проверки адекватности предложенной модели в системе GILDA собраны данные с 30 марта 2006 года по 2 апреля 2007 года (всего 34 тысячи записей). Данные, полученные в формате XML, преобразованы в формат, пригодный для дальнейших экспериментов. Затем для каждого пользователя данные разбивались на обучающую (85 %) и тестовую (15 %) выборки.

Структурная идентификация модели

В качестве нейросетевой модели выбрана многослойная нейронная сеть прямого распространения информации (персептронного типа) с одним скрытым слоем, обучаемая по методу обратного распространения ошибки. Проведены эксперименты по определению оптимальной размерности скрытого слоя нейронной сети. Оптимальной считалась такая размерность, при которой средний процент правильной классификации поведения пользователей для тестовой выборки всех пользователей был максимальным. В процессе экспериментов оказалось, что оптимальная размерность скрытого слоя составляет 20 нейронов, при которой достигается 85,81 % правильной классификации. При большем количестве нейронов в скрытом слое происходило насыщение и дальнейшее увеличение не позволяло повысить процент правильной классификации.

Для каждой модели пользователя определены весовые коэффициенты и параметры обучения ($\eta = 0{,}3$, $\mu = 0{,}15$). При этом использовался некумулятивный вариант метода обратного распространения ошибки. При таком подходе в процессе обучения присутствует элемент случайности, что позволяет повысить вероятность непопадания в локальный минимум.

Экспериментальная верификация модели

Для проверки эффективности предложенной модели была проведена серия экспериментов. Для того чтобы проверить, насколько нейронная сеть способна отличить поведение одного пользователя от поведения другого, использовалась процедура подмены пользователя. На вход нейронной сети, обученной для одного пользователя (легального), подавались данные друго-

го пользователя (нелегального). Так имитировалась ситуация, когда нелегальный пользователь работает под именем (учетной записью) легального пользователя.

Из рис. 2 видно, что предложенная модель позволяет уверенно обнаружить подмену пользователя, поэтому она достаточно эффективна (ошибка первого рода составляет 0,86 %, второго рода — 0, 7%).

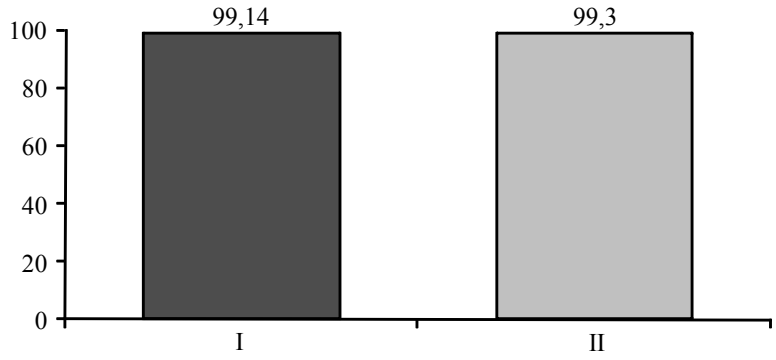

Рис. 2. Процент правильной классификации сеансов для разных пользователей:
I — процент правильной классификации для легального пользователя
для тестовой выборки (соответствует ошибке первого рода);
II — процент правильной классификации для нелегального пользователя
(соответствует ошибке второго рода)

Заключение

В данной статье рассмотрены существующие средства обеспечения безопасности в Grid-системах и подходы к построению моделей поведения пользователей таких систем. Существующие средства обеспечения безопасности в Grid-системах (например, Globus GSI) позволяют обеспечить аутентификацию, авторизацию, обмен сертификатами и использовать ряд других важных подходов к обеспечению безопасности. Вместе с тем средства мониторинга поведения пользователей в Grid-системах недостаточно развиты и не позволяют выявлять их аномальное поведение. Поэтому в данной работе предложен новый подход к анализу поведения пользователей и выявлению аномалий в их работе.

В предложенной модели поведения пользователей Grid-системы учитывается ряд параметров (профиль пользователя), получаемых при выполнении задачи на ресурсах Grid-системы. Для реализации предложенной модели использовалась нейронная сеть прямого распространения, которая функционирует в режиме классификатора. На основе проведенных экспериментов определена оптимальная структура нейросетевой модели, а именно 12-20-1 (20 нейронов в скрытом слое), а также значения весовых коэффициентов и параметров обучения ($\eta = 0{,}3$, $\mu = 0{,}1$).

Для проверки эффективности предложенной модели проведены эксперименты на реальных данных, собранных с помощью средства мониторинга GridICE в Grid-системе GILDA-EGEE. Результаты экспериментов показали, что в 90 % случаев использование модели позволяет обнаружить подмену

пользователя. Таким образом, верификация модели на реальных данных подтвердила эффективность ее применения для выявления аномальной деятельности пользователей в Grid-системах.

1. *Shelestov A.Yu., Kussul N.N., Skakun S.V.* Grid technologies in monitoring systems based on satellite data // J. of Autom. and Inform. Sci. — 2006. — **38**, N 3. — P. 69–80.
2. *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.
3. *Fusco L.* Earth science GRID on demand // Представлено на заседании рабочей группы CEOS WGISS-21 GRID Task Team. — Budapest, Hungary. — May 2006.
4. *Holtman K.* CMS requirements for the Grid // Proc. of the Int. Conf. on Computing in High Energy and Nuclear Physics (CHEP2001). — 2001.
5. *Peltier S.T. et al.* The telescience portal for advanced tomography applications // J. of Parallel and Distributed Comput.: Comput. Grid. — 2002. — **63**, N 5. — P. 539–550.
6. *Annis J., Zhao Y. et al*. Applying chimera virtual data concepts to cluster finding in the sloan sky survey // Techn. Rep. GriPhyN-2002-05, 2009. — 54 p.
7. *Foster, I., Kesselman, C., Tuecke, S.* The Anatomy of the Grid: enabling scalable virtual organizations // Int. J. Supercomput. Appl. — 2001. — **15**, N 3. — 25 p.
8. *Cornwall L.A., Jensen J., Kelsey D.P. et al*. Authentication and authorization mechanisms for multi-domain Grid environments // J. of Grid Comput. — 2004. — N 9. — P. 301–311.
9. *Рамакришнан Л.* Защита Grid // Открытые системы. — 2004. — № 06. — С. 63–68.
10. *Foster I., Kesselman C., Tsudik G., Tuecke S.* A security architecture for computational Grids // ACM Conf. on Comput. and Security. — 1998. — P. 83–91.
11. *IETF*, Public-Key Infrastructure (X.509). — http://www/ietf/org/html/charters.
12. *Adams C., Lloyd S.* Understanding PKI: concepts, standards, and deployment considerations, 2nd ed. — N.Y.: Addison-Wesley, 2000. — 352 p.
13. *GridICE*. — http://gridice.forge.cnaf.infn.it.
14. *MOGAS*. — http://ntu-cg.ntu.edu.sg/pragma/index.jsp.
15. Internet X.509 public key infrastructure proxy certificate profile / S. Tuecke, D. Engert, I. Foster, M. Thompson, L. Pearlman, C. Kesselman // IETF, Draft draft-ietfpkix-proxy-01.txt, 2001.
16. *Simple* Object Access Protocol (SOAP) 1.1. W3C, Note 8, 2000.
17. *Workflow-based* authorization service in the Grid / K. Seung-Hyun, H.K. Kyong, K. Jong, H. Sung-Je, K. Sangwan // J. of Grid Comput. — 2004. — N 2. — P. 43–55.
18. *Shingo T., Susumu D., Shinji S.* A user-oriented secure filesystem on the Grid // The 3rd IEEE/ACM Int. Symp. on Cluster Comput. and the Grid (CCGrid 2003). — May, 2003. — P. 139–143.
19. *Куссуль Н.Н., Скакун С.В.* Нейросетевая модель пользователей компьютерных систем // Кибернетика и вычисл. техника. — 2004. — Вып. 143. — С. 55–68.
20. *Скакун С.В.* Непараметрическая идентификация комплексной нейросетевой модели поведения пользователей компьютерных систем // Там же. — 2005. — Вып. 147. — С. 45–69.
21. *Haykin S.* Neural networks: a comprehensive foundation. — Upper Saddle River, New Jersey: Prentice Hall, 1999. — 1104 p.

Институт космических исследований
НАН Украины и НКА Украины, Киев,
ООО «Интеграция-Плюс», Киев

Получено 25.12.2007