

АНУФРИЕВ Владимир Натанович

ИНТЕГРАЦИЯ PHP5 И IBM DOMINO 7

(редакция 2)

**Москва
2009**

Vladimir Anufriev
PHP5 and IBM Domino 7 Integration

Second Edition (November 2009)

Comments may be addressed to: avn@avnsite.com

When you send information to AVN, you grant AVN a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright by Vladimir Anufriev 2009. All rights reserved.

Введение

Сервер приложений Domino¹ компании IBM² включает в себя HTTP/1.1-совместимый сервер с поддержкой SSL, инструментальные средства разработки приложений (HTML, XML, Lotus Script, @-формулы, Java, JavaScript) на основе GUI и позволяет на одной вычислительной установке достаточной производительности поддерживать протоколы NOTES (почта и сервер приложений), SMTP, POP3, IMAP, LDAP, HTTP, а также их варианты со встроенной криптографической защитой. Такое совмещение функций позволяет максимально снизить издержки на системное администрирование и содержание аппаратного парка при достаточной масштабируемости для предприятия среднего размера³.

Значительная часть готовых свободно распространяющихся приложений, популярных в корпоративной среде, написана с использованием технологии PHP⁴ (от англ. PHP: Hypertext Preprocessor). В частности, консоль управления Base⁵ для системы обнаружения вторжений Snort⁶ написана с применением этой технологии. Поэтому актуальность интеграции уже развернутого сервера приложений Domino с технологиями, основанными на PHP не оставляет никаких сомнений.

В заключении мы покажем, что сервер Domino легко интегрируется с любыми приложениями, основанными на вызове CGI-процедур (от англ. *Common Gateway Interface* — «общий интерфейс шлюза»). Единственной проблемой, которую необходимо решить, является применение аутентификационных и авторизационных механизмов Domino в комплексном решении.

Это вторая редакция данной статьи. Мы уточнили описание параметров настройки PHP, также хотим сообщить, что PHP не работает с версией Domino 8.5. Согласно непроверенным нами пока сообщениям, необходимо обновление до версии 8.5FP1 или 8.5.1.

Интеграционные механизмы

С эталонным HTTP-сервером Apache⁷ PHP интегрируется двумя способами — через вызов определенной CGI-процедуры или через специальный интегрированный модуль, входящий в состав дистрибутива PHP и представляющий собой динамически загружаемую библиотеку функций.

В стек включенного в Domino HTTP-сервера IBM встроила так называемые *внешние фильтры* (DSAPI filter files), которые позволяют разработчикам применять собственные приложения на языке Си для изменения стандартных результатов поэтапной обработки HTTP-запросов или подмены встроенных обработчиков. Приложения пишутся с использованием интерфейса DSAPI и оформляются в виде библиотек функций с поздним

1 <http://www-142.ibm.com/software/sw-lotus/products/product4.nsf/wdocs/dominohomepage>

2 <http://www.ibm.com>

3 Один x86-сервер стоимостью до \$20000 в масштабе цен и аппаратных возможностей 2001 года, сопровождаемый одним системным программистом, на практике позволяет обслуживать до 2000 пользователей в корпоративной сети.

4 Официальный сайт <http://www.php.net>. Историческую справку и краткое описание технологии на русском языке можно прочесть в интернет-энциклопедии <http://ru.wikipedia.org/wiki/PHP>

5 Официальный сайт проекта Basic Analysis and Security Engine (BASE) <http://base.secureideas.net/>

6 Официальный сайт <http://www.snort.org/>

7 Официальный сайт <http://httpd.apache.org/>

связыванием (dynamic link libraries). DSAPI входит в состав Notes C API, который может быть загружен с сайта IBM⁸.

Однако для применения данной технологии для интеграции с PHP нет готового решения. Кроме того, как было показано в предыдущей статье⁹, логика, заложенная в DSAPI, делает написание сложных расширений затруднительным.

В низко и средненагруженных приложениях вполне можно остановиться на вызове PHP как CGI-процедуры.

Интегрированный в сервер приложений HTTP-сервер с самого своего появления позволяет вызывать CGI-процедуры, а также имеет встроенный механизм авторизации доступа к файлам.

Установка и настройка PHP-приложения на примере Base

Все описанные ниже действия проводятся в среде Linux (дистрибутив Suse Linux 10.0), в которой также развернут сервер приложений Domino.

Установим PHP версии 5¹⁰ в каталог /opt/php. Поскольку в наш дистрибутив входит не самая последняя версия, а также в связи с необходимостью подключения к СУБД PostgreSQL¹¹, мы собрали последнюю имевшуюся версию из исходных текстов. Процедура сборки и установки PHP многократно описана в Интернете на различных языках, поэтому не будем на ней останавливаться подробно, дадим только пару замечаний.

В файле /opt/php/lib/php.ini не забудем указать пути к нашему приложению:

```
;;;;;;;;;;;;;
; Paths and Directories ;
;;;;;;;;;;;;;

; UNIX: "/path1:/path2"
include_path = "/opt/php:/opt/php/lib/php:/opt/php/lib/php/Graph"
```

В данную редакцию статьи мы внесли незначительные коррективы, которые уточняют настройки PHP.

В материалах IBM указывается, что нужно обязательно задать значение параметра cgi.force_redirect. По-умолчанию, он установлен в 1. Для работы PHP под Domino нужно установить его в 0:

```
; cgi.force_redirect is necessary to provide security running PHP as a CGI under
; most web servers. Left undefined, PHP turns this on by default. You can
; turn it off here AT YOUR OWN RISK
; **You CAN safely turn this off for IIS, in fact, you MUST.**
cgi.force_redirect = 0
```

Base, начиная с версии 1.4.1, использует Mail.php, поэтому его нужно не забыть

8 <http://www-128.ibm.com/developerworks/lotus/downloads/toolkits.html> – Notes C Api

9 В.Ануфриев. Интеграция Tomcat 5.0 и Lotus Domino 6.5

<http://www.avnsite.com/downloads/domino/Domino2Tomcat.pdf>

10 Текущая версия 5.2.6 собрана нами из исходных текстов с нижеуказанными параметрами конфигурации:

```
./configure \
--with-pgsql \
--prefix=/opt/php \
--enable-fastcgi \
--with-gd
```

11 Официальный сайт <http://www.postgresql.org/>

Мы используем сборку собственного варианта, см. <http://www.avnsite.com/downloads/1c/index.jsp>

установить. Это делается командами:

```
# cd /opt/php/bin
# ./pear install Mail
...
# ./pear install Mail_mime
...
```

Развернем дистрибутив Base в каталог /opt/base-1.4.1.

Поскольку сервер приложений Domino работает под полномочиями учетной записи **notes** (это стандартное значение, предлагаемое по-умолчанию при установке Domino), то необходимо обеспечить доступ этой учетной записи к каталогу /opt/php на чтение, а к каталогу /opt/base-1.4.1 на запись. Приложение Base при первом запуске создает в своем корневом каталоге файл base_conf.php с параметрами настройки. Теоретически, можно этот файл создать вручную по шаблону base_conf.php.dist или после того, как он создается, закрыть доступ к каталогу на запись с целью повышения безопасности.

Если установка идет под полномочиями root, то команды будут приблизительно следующие:

```
# chmod -R o+r /opt/php
# chown root:notes /opt/base-1.4.0
# chmod g+w /opt/base-1.4.0
```

Base использует ADOdb Database Abstraction Library for PHP¹², которую развернем в каталог /opt/adodb5.

Теперь запустим Domino Administrator и создадим сайт с адресом ids.avnsite.com. Здесь нужно отметить, что сайт, защищенный протоколом SSL, в сервере Domino привязывается к IP-адресу, т.е. каждый защищенный виртуальный сервер должен быть связан с как минимум одним IP-адресом, а один IP-адрес нельзя назначить на несколько виртуальных серверов. Мы не указываем IP-адреса в синонимах сервера, поэтому наш сайт будет доступен только по протоколу HTTP (порт 80). В разделе Configuration:Internet Sites нажмем кнопку Add internet site и выберем тип Web (см. рис. 1).

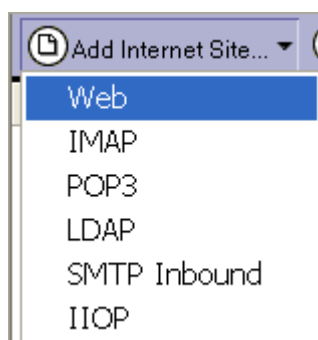


Рис. 1. Переход к конфигурированию интернет сайта из меню Domino Administrator

Для создания сайта необходимо заполнить несколько страниц закладок в открывшемся окне конфигурации. На первой закладке мы заполняем основные параметры, которые подробно описаны в документации Domino и не требуют дополнительных пояснений (см. рис. 2).

12 Официальный сайт <http://adodb.sourceforge.net/>



Web Site IDS

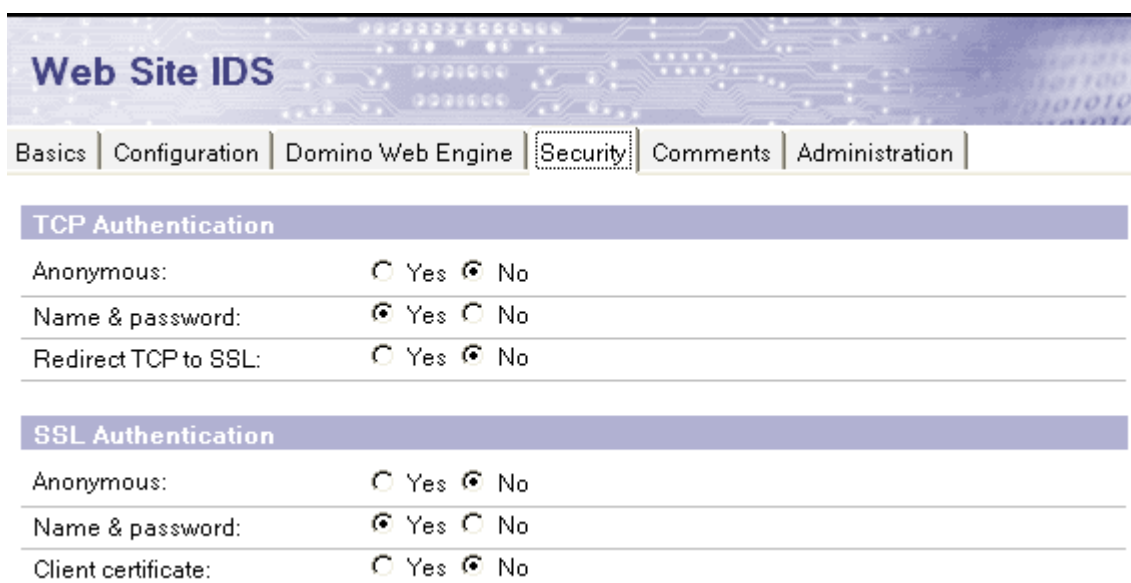
Basics | Configuration | Domino Web Engine | Security | Comments | Administration

Site Information

Descriptive name for this site:	IDS
Organization:	AVN Holding Ltd.
Use this web site to handle requests which cannot be mapped to any other web sites:	☐ Yes ☒ No Note: only one web site should have this option set to Yes
Host names or addresses mapped to this site:	ids.avnsite.com
Domino servers that host this site:	*

Рис. 2. Страница Basics настройки интернет сайта

На закладке Security мы настраиваем, фактически, только одну опцию. Для того, чтобы происходил только авторизованный доступ к ресурсам необходимо запретить анонимный доступ и разрешить аутентификацию по имени и паролю (см. рис 3).



Web Site IDS

Basics | Configuration | Domino Web Engine | Security | Comments | Administration

TCP Authentication

Anonymous:	☐ Yes ☒ No
Name & password:	☒ Yes ☐ No
Redirect TCP to SSL:	☐ Yes ☒ No

SSL Authentication

Anonymous:	☐ Yes ☒ No
Name & password:	☒ Yes ☐ No
Client certificate:	☐ Yes ☒ No

Рис. 3. Страница Security настройки интернет сайта

Заполнение опций на закладке Domino Web Engine мы описывать не будем, так как это не касается напрямую темы данной статьи. На ней можно все оставить по-умолчанию, если вы не используете установку с несколькими серверами и механизмом Single-Sign-On. А вот на закладке Configuration остановимся подробнее (см. рис. 4).

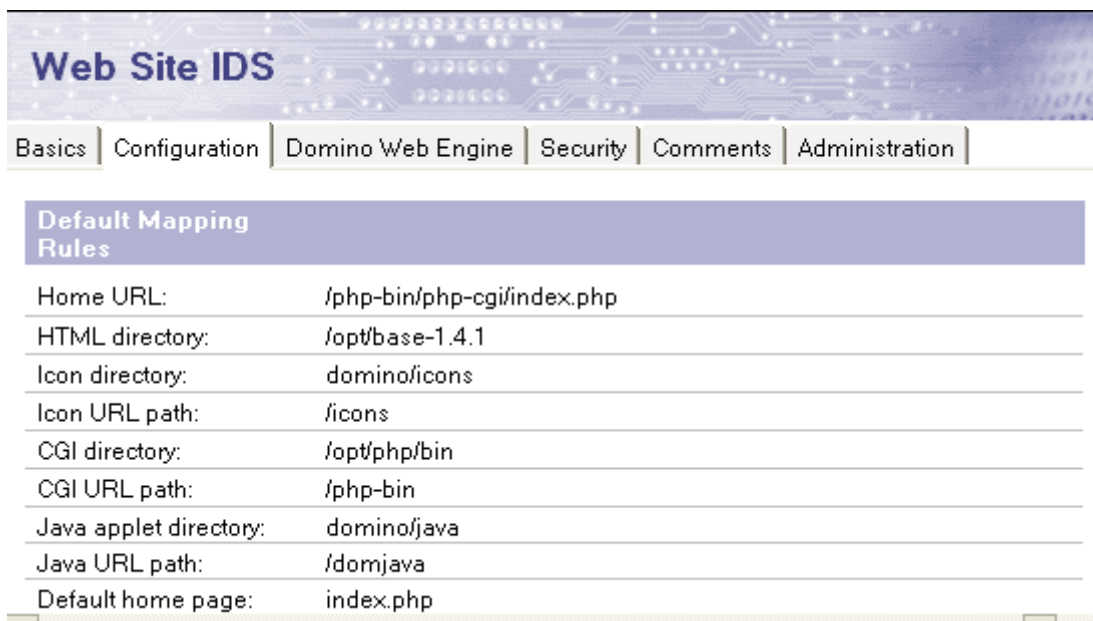


Рис. 4. Страница Configuration настройки интернет сайта

Здесь обязательно нужно заполнить поле Home URL. И этого оказывается мало для корректной работы сайта, придется корректировать исходные тексты на PHP, чтобы все работало, как привычно. В поле HTML directory мы указываем путь к установленному приложению Base. В этом каталоге и его подкаталогах лежат модули приложения на PHP и вспомогательные ресурсы. Пара следующих полей в нашем приложении не используется. В поле CGI directory мы должны указать каталог, в котором располагаются CGI-процедуры, в нашем случае достаточно и одной такой процедуры, которая называется php-cgi. В поле CGI URL path укажем префикс URL, который будет мапироваться на /opt/php/bin. Таким образом, когда в URL запроса будет встречаться шаблон /php-bin, сервер будет интерпретировать его как необходимость поиска в каталоге /opt/php/bin. Остальные поля закладки Configuration в нашем случае несущественны.

Теперь нам нужно разграничить доступ к нашему приложению. Это можно сделать через меню (см. рис. 5).

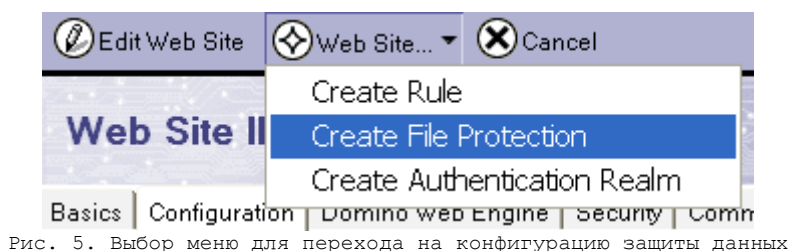


Рис. 5. Выбор меню для перехода на конфигурацию защиты данных

Воспользовавшись документацией (Lotus Domino Administrator 7 Help), необходимо ограничить доступ к нашим CGI-процедурам (см. рис. 6).

Save & Close Cancel

Web Site File Protection

Basics | Comments | Administration

Basics

Description: PHP

Directory or file path: /opt/php/bin/*

Access Control

Current access control list: Владимир Н Ануфриев/Groupe5/FR -(POST and GET) -Default- -(No Access)

Set/Modify Access Control List

Рис. 6. Ограничение доступа к каталогу с CGI-процедурами

Поскольку напрямую мы к модулям приложения Base не обращаемся (это делает CGI-процедура /opt/php/bin/php-cgi), то, на всякий случай, запретим обращение браузера к каталогу /opt/base-1.4.1. В результате наша конфигурация в основном окне IBM Domino Administrator будет выглядеть как на рисунке 7.

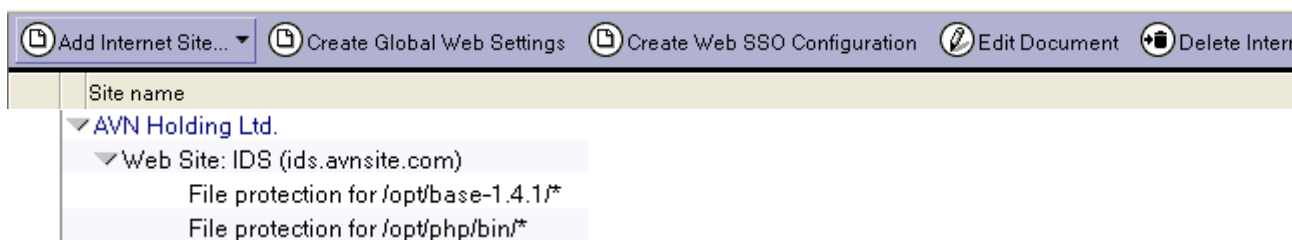


Рис. 7. Вид с конфигурацией нашего сервера

Поскольку наши изменения привели к созданию нового виртуального сервера, то для вступления их в силу необходимо перезагрузить HTTP-сервер.

После перезапуска командами `tell http quit`, `load http` на консоли сервера мы увидим приблизительно следующее:

```
[06120:00009-1178098608] load http
[06120:00100-1189239728] 08.10.2008 05:45:36 PM Remote console command issued by Владимир Н Ануфриев/Groupe5/FR: load http
[14295:00002-1120477312] 08.10.2008 05:45:39 PM HTTP Server: Using Internet Site Configuration View
[14295:00002-1120477312] 08.10.2008 05:45:40 PM JVM: Java Virtual Machine initialized.
[14295:00002-1120477312] 08.10.2008 05:45:40 PM HTTP Server: Java Virtual Machine loaded
[14295:00002-1120477312] 08.10.2008 05:45:40 PM Attempting to start Tomcat: /usr/share/tomcat5/bin/startup.sh
[14295:00002-1120477312] 08.10.2008 05:45:40 PM Apache Tomcat Connector (Jakarta/Connector 1.2.6/DSAPI/1.13.1) loaded
[14295:00002-1120477312] 08.10.2008 05:45:40 PM HTTP Server: DSAPI Apache Tomcat Interceptor (Jakarta/Connector 1.2.6/DSAPI/1.13.1)
Loaded successfully
[14295:00002-1120477312] 08.10.2008 05:45:40 PM Servlet engine initialization was successful
[14295:00002-1120477312] 08.10.2008 05:45:45 PM HTTP Server: Started
```

Если мы в дальнейшем будем менять только настройки прав доступа, то для обновления конфигурации достаточно дать серверу команду `tell http refresh`.

С помощью приложения Domino Web Server Configuration (7) мы назначили нашему сайту созданную ранее средствами Domino Designer нестандартную форму для авторизации на сервере Domino. Поэтому страница в браузере при обращении к нашему сайту будет выглядеть совсем необычно (см. рис. 8).

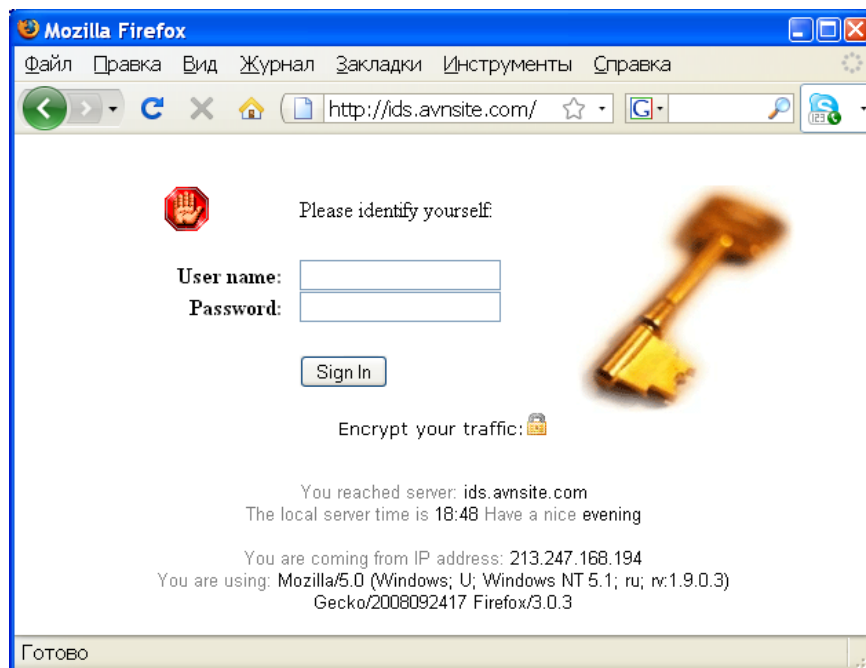


Рис. 8. Форма для авторизации

После ввода имени и пароля заданного пользователя мы попадаем на окно настройки приложения Base (см. рис. 9).

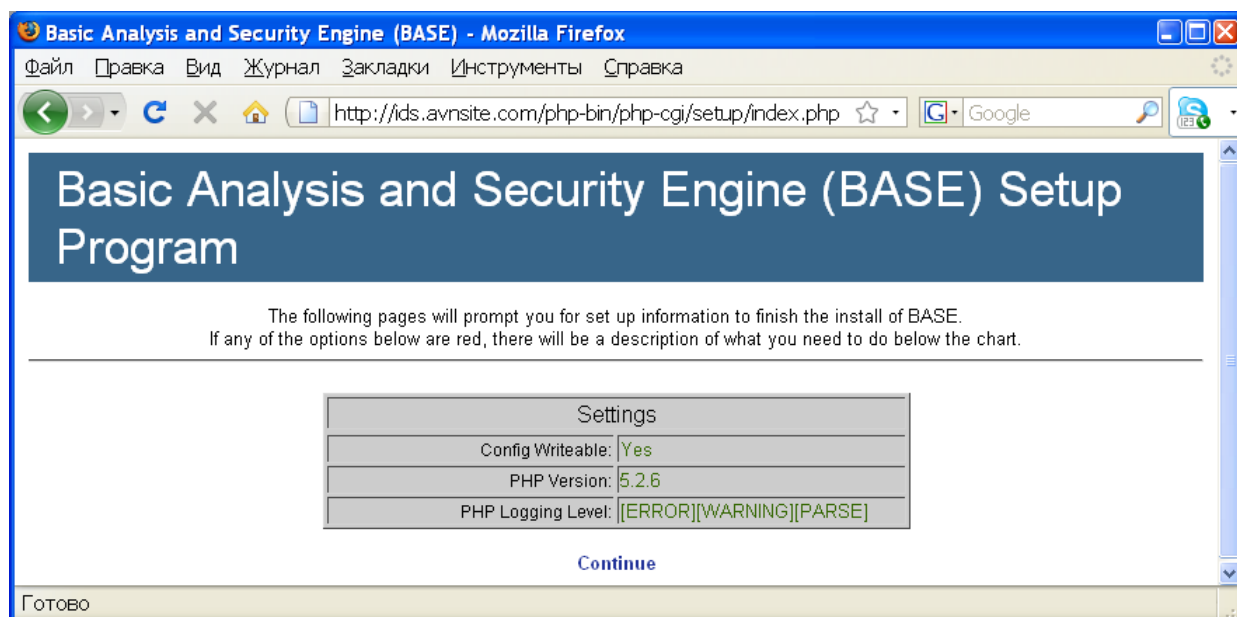


Рис. 9. Форма для настройки приложения Base

Дальше нужно пройти несколько страниц несложных настроек. И если мы все сделали правильно, то в конце концов мы увидим главную страницу работающего приложения, представленную на рисунке 10.

Настройка произвольного CGI-приложения на примере Nagios

Настройка произвольного приложения, основанного на технологии CGI, мало чем отличается от настройки приложения PHP. Описав соответствующий сайт, ограничиваем доступ к используемым CGI-процедурам и ресурсам. В завершении процесса по возможности включаем в меню приложения ссылку на процедуру завершения сессии с сервером приложений.

Развернем приложение Nagios¹³ для мониторинга сетевых проблем в каталог /usr/local/nagios.

Создадим сайт nagios.avnsite.com аналогично тому, как мы делали в предыдущем параграфе (см. рис. 11).

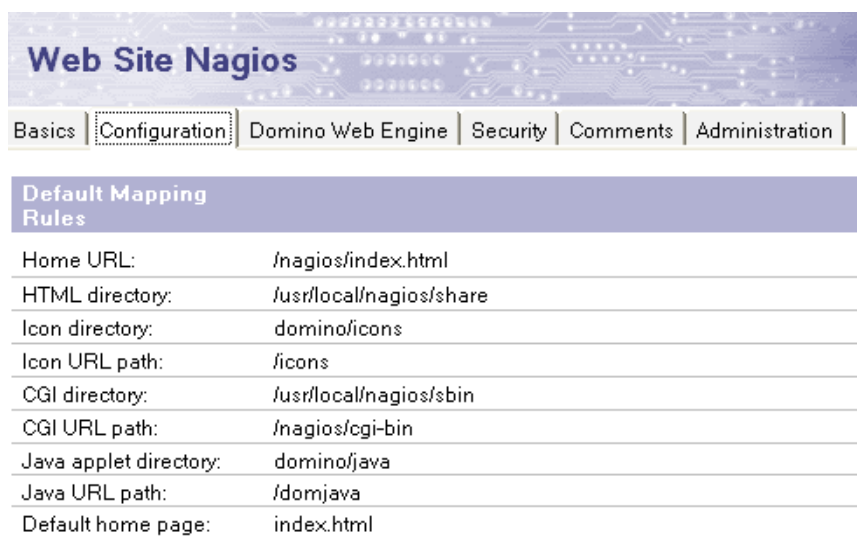


Рис. 11. Закладка Configuration настройки сайта приложения Nagios

Поскольку в данном случае доступ к HTML-страницам получает непосредственно браузер, а не как в случае с Base только CGI-процедура, то необходимо указать, откуда брать страницы, в случае наличия префикса /nagios в URI (см. рис. 12).



Рис. 12. Правило подстановки при поиске файла по указанному префиксу URI

В результате общая конфигурация сервера будет выглядеть как на рисунке 13.

13 Официальный сайт <http://www.nagios.org/>

```

▼ Web Site: Nagios (nagios.ag-loyalty.com; nagios.avnsite.com;
nagios.local.ag-loyalty.com; nagios.local.avnsite.com; 192.168.1.4; 192.168.1.3)
File protection for /usr/local/nagios/sbin/*
File protection for /usr/local/nagios/share/*
Rule (directory with Read access): /nagios -> /usr/local/nagios/share

```

Рис. 13. Общий вид конфигурации сервера

Заметим, что в данном случае из локальной сети мы можем обращаться к серверу по защищенному протоколу HTTPS, так как в конфигурации сервера указаны IP-адреса.

Остается внести последний штрих в меню приложения:

```

--- /usr/local/nagios/share/side.html 2008-04-10 16:43:41.000000000 +0400
+++ /usr/local/nagios/share/side.html 2008-10-08 20:58:43.000000000 +0400
@@ -3,6 +3,7 @@
<HTML>
<HEAD>
<META NAME="ROBOTS" CONTENT="NOINDEX, NOFOLLOW">
+<META http-equiv="Content-Type" content="text/html; charset=UTF-8">
<TITLE>Nagios</TITLE>

<STYLE type="text/css">
@@ -271,6 +272,10 @@
    <td width=13></td>
    <td nowrap width=134><a href="/nagios/cgi-bin/config.cgi" target="main" onMouseOver="switchdot('config-dot',1)"
onMouseOut="switchdot('config-dot',0)" c
    </tr>
+   <tr>
+   <td width=13></td>
+   <td nowrap width=134><a href="/names.nsf?Logout&RedirectTo=/" target="main" onMouseOver="switchdot('config-dot',1)"
onMouseOut="switchdot('config-dot',0
+   </tr>
+ </table>

</BODY>

```

И можем пользоваться!

Сложный случай

По многочисленным просьбам трудящихся электронного фронта мы добавили этот раздел. Если одному приложению необходимо поддерживать несколько каталогов с CGI-процедурами, то отображения с помощью дополнительных правил, например (путь должен указывать на вашу установку PHP, в данном примере он отличается от рассмотренного ранее и соответствует стандартной версии PHP 5.2.11 из поставки Linux OpenSuSe 11.1):

```

Rule (directory with Execute access): /base/* -> /usr/bin/php-cgi/opt/base-1.4.3/*

```

будет недостаточно, а указывать в вышеприведенном правиле *.php будет вообще неправильно, несмотря ни на какие рекомендации от самой IBM (<http://www-01.ibm.com/support/docview.wss?uid=swg21099845>).

Механизм более-менее работает, как описано ранее, только при задании соответствующих путей в основном документе «Web Site». И то при этом приходится указывать дополнительное правило, пример которого приведен на рисунке 14, так как где-то по дороге теряется заголовок **Content-Type**.

Web Site Rule

Basics

Comments

Administration

Basics

Description:	HTTP response headers		
Type of rule:	HTTP response headers		
Incoming URL pattern:	/*.php		
HTTP response codes:	200, 206		
Expires header:	☒ Don't add header ☐ Add header only if application did not ☐ Always add header (override application's header)		
Custom headers:	Name: Content-Type Value: text/html ☒ Override Name: Value: ☐ Override Name: Value: ☐ Override		

Рис. 14. Документ Rule (headers): /.php 200,206

Если посмотреть, какие CGI-переменные будут в этом случае переданы приложению, то в нашем последнем примере с добавленным документом для обработки PHP мы увидим следующее (несущественное для дальнейшего рассмотрения мы отбросили):

```

AUTHENTICATED="YES"
AUTH_TYPE="Domino"
CGI_PATH_INFO="/opt/base-1.4.3/base_main.php"
DOCUMENT_NAME="/usr/bin/php-cgi/opt/base-1.4.3/base_main.php"
DOCUMENT_ROOT="/usr/local/nagios/share/"
DOCUMENT_URI="/base/base_main.php"
GATEWAY_INTERFACE="CGI/1.1"
HTTPS="OFF"
HTTP_ACCEPT="text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8"
HTTP_ACCEPT_CHARSET="windows-1251,utf-8;q=0.7,*;q=0.7"
HTTP_ACCEPT_ENCODING="gzip,deflate"
HTTP_ACCEPT_LANGUAGE="ru"
HTTP_CONNECTION="keep-alive"
HTTP_COOKIE="PHPSESSID=j5shi99676b0c8qocrhl9b5v221tn2qi;DomAuthSessId=652826175187B164C6DDFEF99AD8EAD7"
HTTP_HOST="192.168.1.4"
HTTP_KEEP_ALIVE="300"
HTTP_USER_AGENT="Mozilla/5.0 (Windows; U; Windows NT 5.1; ru; rv:1.9.1.4) Gecko/20091016 Firefox/3.5.4 (.NET CLR 3.5.30729)"
PATH_INFO="/opt/base-1.4.3/base_main.php"
PATH_TRANSLATED="/usr/local/nagios/share/opt/base-1.4.3/base_main.php"
REQUEST_METHOD="GET"
SCRIPT_NAME="/base/base_main.php"

```

Как видим, с путями у нас сплошная путаница. PHP не сможет разобраться в предоставленной информации и выдаст ошибку File not found.

Если бы мы дописали в наше правило в конце *.php, то у значений всех переменных, оканчивающихся на .php, в конце появился бы '/. Здесь мы видим, что правильное значение пути к интерпретируемому скрипту передается только в переменной CGI_PATH_INFO,

которая не входит в стандарт CGI/1.1 и не обрабатывается PHP.

Самое простое решение проблемы, которое можно предложить, состоит в вызове не непосредственно программы `php-cgi`, а некоторой программы, которая поменяет значения переменных как нужно, а затем вызовет `php-cgi`. Поскольку в поставке OpenSuSe 11.1, которую мы используем, реальная процедура PHP называется `/usr/bin/php-cgi5`, то, не изменяя настроек HTTP-сервера, заменим ссылку `/usr/bin/php-cgi` на скрипт, приблизительно следующего содержания:

```
#!/bin/bash

if [ -n "$CGI_PATH_INFO" ] ; then
    export PATH_TRANSLATED=$CGI_PATH_INFO
    export DOCUMENT_ROOT=/
    export DOCUMENT_NAME=$CGI_PATH_INFO
    export SCRIPT_NAME=$CGI_PATH_INFO
    export REQUEST_URI=$DOCUMENT_URI
    /usr/bin/php-cgi5
    if [ $? -eq 255 ] ; then
        echo $CGI_PATH_INFO
        echo `export | sed "s/\$/\r\n/"`
    fi
else
    echo Status: 404 Not found
    echo X-Powered-By: PHP/5.2.11
    echo Content-type: text/html
    echo
    echo "<html>"
    echo "<body>"
    echo Script not specified
    echo "</body>"
    echo "</html>"
fi
```

Теперь PHP сможет работать одновременно с другой группой CGI-приложений.

Заключение

Таким образом, с помощью стандартных настроек сервер приложений Domino может эффективно управлять произвольными CGI-приложениями, в том числе построенными на технологии PHP. При этом прозрачным образом используются аутентификационные и авторизационные механизмы сервера приложений Lotus Domino.

2009-11-05

