

Разворачивание инфраструктуры CRM-системы SugarCRM для управляющей организации (товарищества собственников жилья)

Чистяков Алексей Сергеевич

*Российский экономический университет имени Г. В. Плеханова
студент*

Аннотация

В статье рассматривается пример разворачивания инфраструктуры CRM-системы для малых организаций на основе бесплатной CRM-системы с открытым исходным кодом SugarCRM. В инфраструктуру включены: сервер приложения, сервер базы данных и сервер-шлюз.

Ключевые слова: CRM-система, SugarCRM, разворачивание инфраструктуры, PHP, MySQL, Apache2, OpenVPN

Infrastructure deployment of CRM-system SugarCRM for the management company (homeowners)

Chistjakov Aleksey Sergeevich

*Plekhanov Russian University of Economics
student*

Abstract

The article describes an example of deployment of CRM-system infrastructure for small organizations on the basis of a free open-source CRM-system SugarCRM. The infrastructure includes: an application server, a database server and a gateway server.

Keywords: CRM-system, SugarCRM, infrastructure deployment, PHP, MySQL, Apache2, OpenVPN

Вводная часть

В настоящее время для предприятия любого размера крайне важно идти в ногу со временем и внедрять современные практики ведения бизнеса. Одна из таких практик – принятие модели взаимодействия CRM, основанной на постулате, что центром философии бизнеса является клиент, а главные направления деятельности организации – это принятие мер по повышению эффективности обслуживания клиентов, обеспечения роста уровня продаж и улучшения качества проводимых маркетинговых кампаний. Эта модель подразумевает использование в подавляющем количестве бизнес-процессов компании программного обеспечения, предназначенного для автоматизации стратегий взаимодействия с клиентами и заказчиками. Это программное обеспечение – система управления взаимоотношения с клиентами – называется CRM-системой. Сейчас на рынке CRM-систем представлено

большое количество игроков, предлагающих продукты самого разного уровня. При этом, при анализе этих продуктов прослеживается определенный тренд – уровень возможной модификации системы под нужды конкретной компании прямо пропорционален ее стоимости. Таким образом, небольшие организации, которым, в силу рода их деятельности, нужно внесение определенных модификаций в CRM-систему, сталкиваются с серьезной проблемой – они не могут найти систему, которую можно подстроить под нужды организации, такую, чтобы стоимость ее внедрения соответствовала ограниченному бюджету компании. Именно по этой причине в статье будет рассмотрено развертывание инфраструктуры на основе бесплатной версии (community edition) CRM-системы SugarCRM, подходящей для небольшой организации (до 40 работников).

К таким организациям, в частности, могут относиться управляющие компании (товарищества собственников жилья) для управления многоквартирными домами. В настоящий момент такие организации используют большое количество информационных систем, которые можно отнести к пяти классам. В соответствии с [1], к четвертому классу информационных систем относятся облачные технологии, а к перспективному пятому классу - информационные системы, которые будут использовать технологии Интернета Вещей. Но, несмотря на явный тренд развития степени использования информационных систем в ЖКХ, анализ эксплуатируемых в РФ информационных систем для управления ЖКХ показывает, что остается проблема создания единого информационного пространства для отображения взаимодействия с собственниками жилья. Даже самые совершенные системы для управления ЖКХ, эксплуатируемые управляющими компаниями (товариществами собственников жилья) обычно ориентированы на технических сотрудников управляющих компаний. Это означает, что собственники жилья (в многоквартирных домах) не включены контур управления многоквартирными домами, в которых они непосредственно проживают. При этом такую возможность достаточно давно предоставляют зарубежные информационные системы для управления недвижимостью [2]. Одним из возможных путей формирования единого информационного пространства для собственников жилья может стать интеграция эксплуатируемых в управляющих компаниях (товариществах собственников жилья) информационных систем с CRM-системами. Информационная система SugarCRM является вполне конкурентоспособной для того, чтобы быть выбранной для конкретной управляющей компании (товарищества собственников жилья). При этом выбор будет производиться с помощью алгоритма, который будет учитывать уровень готовности организации к автоматизации [3].

Описание инфраструктуры

Инфраструктура будет состоять из трех серверов: сервера базы данных (в качестве базы данных будет выступать MySQL), сервера приложения, на котором будет расположен веб-сервер Apache2 и непосредственно файлы

SugarCRM и сервера-шлюза, через который можно будет получать доступ к первым двум серверам, кроме того, этот сервер будет использоваться как OpenVPN-сервер для подключения внутренних пользователей CRM-системы компании с целью их работы в этой системе. Таким образом, можно утверждать, что устанавливаемая информационная система не будет предъявлять значительных требований к ИТ-инфраструктуре управляющей организации (ТСЖ).

В качестве серверов будут выступать VDS (виртуальные сервера), расположенные на базе хостера нижнего ценового сегмента DigitalOcean.

Создание и настройка серверов

В панели управления DigitalOcean создадим 3 сервера и дадим им понятные названия, чтобы в дальнейшем с легкостью их различать. Все 3 сервера будут работать под управлением Linux Ubuntu последней версии.

Через несколько минут сервера готовы к работе. Каждый сервер имеет два сетевых интерфейса – WAN («внешний», через который происходит общение сервера с другими ресурсами сети Интернет) и LAN («локальный», через который три созданные сервера общаются между собой).




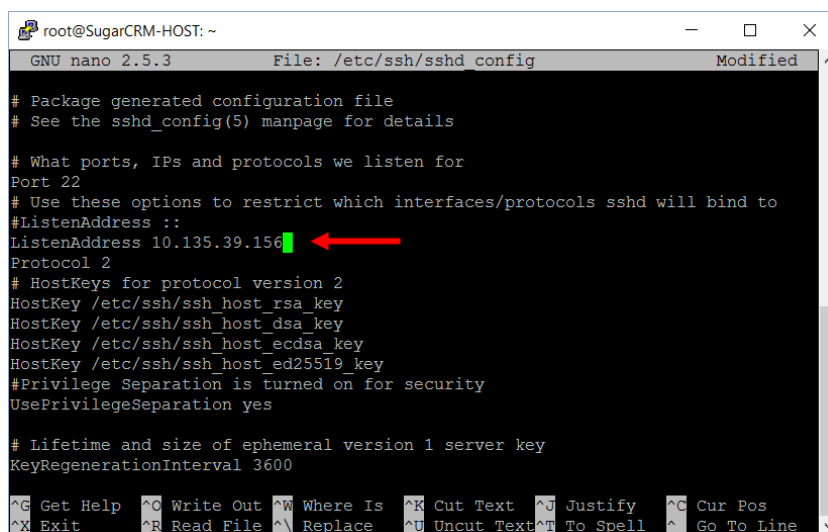
| Name | IP Address |
|---|---------------|
|  SugarCRM-DB 512 MB / 20 GB Disk / FRA1 - Ubuntu 16.04.1 x32 | 138.68.89.161 |
|  SugarCRM-HOST 512 MB / 20 GB Disk / FRA1 - Ubuntu 16.04.1 x32 | 138.68.89.160 |
|  SugarCRM-GW 512 MB / 20 GB Disk / FRA1 - Ubuntu 16.04.1 x32 | 138.68.89.159 |

Рисунок 1 – Созданные сервера и назначенные им IP-адреса WAN-интерфейсов

Добавим сервера в программу PuTTY и сохраним подключения. Поскольку панель управления DigitalOcean позволяет включить авторизацию SSH по ключу для сервера еще на этапе его создания, то процесс перехода с авторизации по паролю на авторизацию по ключу будет опущен. Тем не менее, важно отметить процесс ограничения возможности авторизации по SSH для сервера БД и сервера приложения – на момент создания SSH-процессы всех трех серверов «слушают» IP-адреса, привязанные к WAN-интерфейсам, иными словами – инициировать SSH-подключение может любой пользователь сети Интернет. Это ожидаемое поведение сервера-шлюза, но возможность подключения для двух остальных серверов необходимо ограничить. Делается это очень просто – устанавливаем подключение с сервером-шлюзом, внутри этого подключения инициируем еще одно – сначала с сервером базы данных, а затем с сервером приложения, и для каждого проводим операцию редактирования конфигурационного

файла SSH-сервиса – меняем «прослушиваемый» IP-адрес на таковой у LAN-интерфейса каждого сервера соответственно.



```
root@SugarCRM-HOST: ~
GNU nano 2.5.3 File: /etc/ssh/sshd_config Modified
# Package generated configuration file
# See the sshd_config(5) manpage for details

# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
ListenAddress 10.135.39.156
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^\ Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Рисунок 2 – Процесс редактирования конфигурационного файла SSH-сервиса

Настройка сервера-шлюза

Сервер-шлюз выполняет роль связующего моста между сетью Интернет и внутренней инфраструктурой, в нашем случае состоящей из двух серверов. Даже несмотря на кажущуюся избыточность использования шлюза, опускание его использования и использование прямого подключения к серверам с чувствительными данными (таким, как сервер базы данных или сервер приложения) чревато утечкой данных в результате различных атак, вектор которых без использования шлюза резко расширяется: это и уязвимости в стандартных пакетах ОС Ubuntu, и уязвимости непосредственно в CRM-системе.

Для того, чтобы сотрудники компании могли пользоваться ресурсами CRM, настроим OpenVPN-сервер[4]. Для этого, мы воспользуемся продуктом компании-производителя OpenVPN – OpenVPN Access Server. Этот продукт позволяет автоматизировать большинство процессов настройки VPN-сервера, а также имеет удобную страницу для подключения, используемую клиентами VPN-сервера.

Установка OpenVPN Access Server (сокр. AS) ничем не отличается от установки других программных пакетов, поэтому ее описание будет опущено. После установки AS, произведем несколько важных шагов.

Во-первых, отключим маршрутизацию всего трафика клиентов через VPN-сервер, чтобы ресурсы в сети Интернет открывались у пользователей без участия сервера-шлюза. Во-вторых, включим «проброс» в VPN-сеть локальной сети, в которой находятся сервера компании. Стоит отметить, что AS, как правило, автоматически определяет сетевые интерфейсы, имеющие приватные IP-адреса и подставляет их в список сетей, подлежащих «пробросу».

Routing

Should VPN clients have access to private subnets (non-public networks on the server side)?

- No
- Yes, using NAT
- Yes, using routing (advanced)

Specify the private subnets to which all clients should be given access (as 'network/netmask_bits', one per line):

```
10.19.0.0/20
10.135.0.0/16
```

Should client Internet traffic be routed through the VPN?

- No
- Yes

Рисунок 3 – Настройка маршрутизации AS

Наконец, создадим первого пользователя – user1 – и разрешим ему подключение к VPN. После этого перезапустим AS.

| Username | Group | More Settings | Admin | Allow Auto-login | Deny Access | Delete |
|----------|------------------|---------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|
| openvpn | No Default Group | Show | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | |
| user1 | No Default Group | Hide | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

Local Password: (Change Password)

Select IP Addressing: Use Dynamic Use Static

Access Control

Select addressing method: Use NAT Use routing

Allow **Access To** these Networks:

List subnets in *network/nbits* form

all server-side private subnets

all other VPN clients

VPN Gateway

Configure VPN Gateway: No Yes

DMZ settings

Configure DMZ IP address: No Yes

New Username: No Default Group Show

Require user permissions record for VPN access

Рисунок 4 – Добавление нового пользователя в AS

Наконец, зайдём на страницу авторизации пользователя, укажем данные созданного ранее пользователя user1 и выполним подключение.

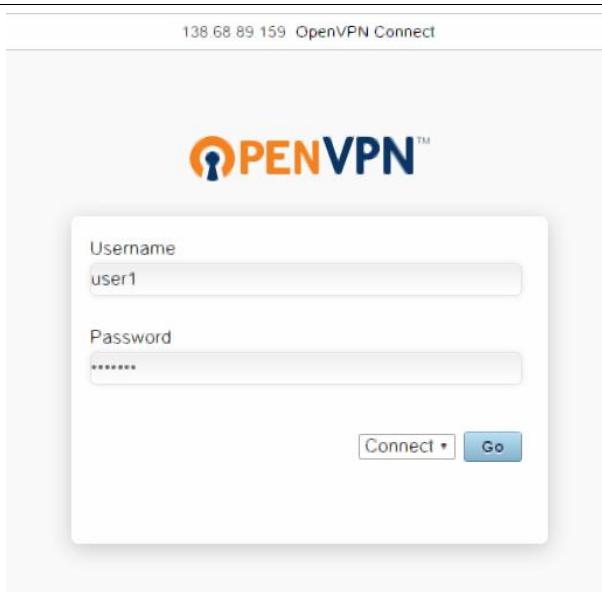


Рисунок 5 – Страница авторизации пользователя AS

На этом настройка шлюза закончена.

Настройка сервера БД

Здесь и далее будем считать, что все компьютеры, с которых происходит обращение к любому из трех серверов, подключены к серверу-шлюзу через OpenVPN AS (если не указано иное).

Подключимся к серверу БД, используя его IP-адрес в локальной сети, а затем установим БД MySQL[5]. Отредактируем файл конфигурации MySQL, разрешив подключение к БД только из локальной сети – для этого установим в качестве «прослушивающего» IP-адреса тот, который назначен для LAN-интерфейса сервера.

```
[mysqld]
#
# * Basic Settings
#
user           = mysql
pid-file       = /var/run/mysqld/mysqld.pid
socket         = /var/run/mysqld/mysqld.sock
port          = 3306
basedir        = /usr
datadir        = /var/lib/mysql
tmpdir         = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address   = 10.135.39.170
#
# * Fine Tuning
#
```

Рисунок 6 – Установка ограничения на подключение к БД

Перезапустим MySQL и подключимся к БД, используя СУБД HeidiSQL. Для этого, в качестве хоста укажем IP-адрес сервера БД в

локальной сети, в качестве имени пользователя – root, в качестве пароля – установленный во время установки MySQL пароль.

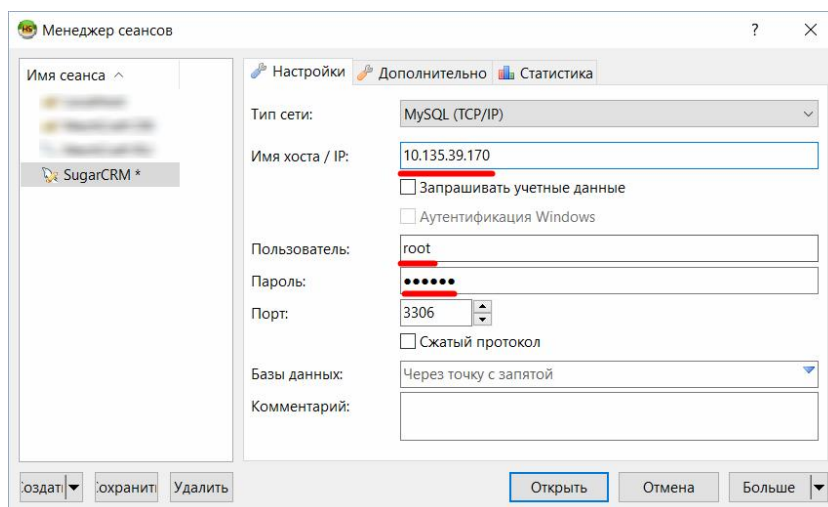


Рисунок 7 – Настройка соединения в СУБД HeidiSQL

Создадим новую базу данных, которая будет использоваться SugarCRM, а также пользователя, которого система будет использовать для подключения.

При создании пользователя, укажем, что подключаться от его имени можно только с IP-адресов локальной сети, а также выдадим все привилегии на созданную ранее БД.

На этом настройка сервера БД закончена.

Настройка сервера приложения

Загрузим zip-архив SugarCRM с репозитория разработчика на GitHub на сервер приложения и распакуем его в папку /var/www/sugarcrm

Установим веб-сервер Apache2 и интерпретатор PHP[5], языка, на котором написана SugarCRM. Установка проста и сводится к введению команды на установку пакетному менеджеру, поэтому рассматриваться здесь не будет. После установки необходимо настроить веб-сервер – установить его «прослушивание» на IP-адрес сервера приложения в локальной сети, чтобы CRM-система не была видна третьим лицам (не подключенным к VPN на шлюзе)[6].

```
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 10.135.39.156:80

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

Рисунок 8– Настройка IP-адреса для «прослушивания» веб-сервером Apache2

Проверим корректность установки веб-сервера: если все сделано верно, то, при наборе в браузере внешнего IP-адреса сервера приложения, должна быть выдана ошибка, а при наборе внутреннего (локального) IP-адреса – должна открываться стандартная проверочная страница.



Рисунок 9 – Стандартная проверочная страница веб-сервера

Чтобы вместо тестовой страницы открывалась непосредственно SugarCRM, изменим файл настроек стандартного хоста (сайта) [7]: укажем в качестве корневой директории директорию `/var/www/sugarcrm`, в которую ранее были распакованы файлы CRM-системы.

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/sugarcrm ←

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn
```

Рисунок 10 – Изменение корневой директории

Наконец, перезагрузим конфигурацию веб-сервера.

На этом, настройка сервера приложения закончена: в браузере открывается страница менеджера установки SugarCRM.

Настройка SugarCRM

Установка SugarCRM проводится с помощью встроенного мастера установки, поэтому большинство шагов не вызовет сложности. Тем не менее, стоит остановиться на некоторых нетривиальных.

Шаг проверки корректности установки проверяет, установлены ли необходимые модули для интерпретатора PHP, а также правильно ли указаны уровни разрешений для файлов и директорий системы. На рисунке ниже показаны примеры ошибок, которые могут возникнуть при проверке.

| | |
|---|---|
| Writable SugarCRM Configuration File (config.php) | The config file exists but is not writeable. Please take the necessary steps to make the file writeable. Depending on your Operating system, this might require you to change the permissions by running <code>chmod 766</code> , or to right click on the filename to access the properties and uncheck the read only option. |
| | The config override file exists but is not writeable. Please take the necessary steps to make the file writeable. Depending on your Operating system, this might require you to change the permissions by running <code>chmod 766</code> , or to right click on the filename to access the properties and uncheck the read only option. |
| Writable Custom Directory | The Custom Directory exists but is not writeable. You may have to change permissions on it (<code>chmod 766</code>) or right click on it and uncheck the read only option, depending on your Operating System. Please take the needed steps to make the file writeable. |
| Writable Cache Sub-Directories | The files or directories listed below are not writeable or are missing and cannot be created. Depending on your Operating System, correcting this may require you to change permissions on the files or parent directory (<code>chmod 755</code>), or to right click on the parent directory and uncheck the 'read only' option and apply it to all subfolders. |

Рисунок 11 – Проверка корректности установки обнаружила несколько ошибок

В этом конкретном случае, понадобилось изменить права доступа для файла конфигурации системы `config.php`, а также установить владельцем всех папок CRM-системы пользователя, от имени которого запущен веб-сервер Apache2 (`www-data`).

Следующий важный шаг – настройка соединения с базой данных. Для этого, в поле IP-адреса введем локальный IP-адрес сервера БД, а в поля учетных данных БД – информацию о созданной ранее базе данных и соответствующему ей пользователю.

http://10.135.39.156/install.php

Help

SUGARCOMMUNITY EDITION.
 SUGARCRM

Database Configuration

* Required field

Provide Database Name

Name of the database that will contain the data for the Sugar instance you are about to install:

* Database Name

Name of web server or machine (host) on which the database is located (such as localhost or www.mydomain.com):

* Host Name

Provide Database User Information

The username and password of a database administrator who can create database tables and users and who can write to the database is necessary in order to set up the Sugar database.

* Database Administrator Username

Database Admin Password

For security purposes, you can specify an exclusive database user to connect to the Sugar database. This user must be able to write, update and retrieve data on the Sugar database that will be created for this instance. This user can be the database administrator specified above, or you can provide new or existing database user information.

Sugar Database Username

Choose Demo Data

Populate Database with Demo Data?

Back
Next

Рисунок 12 – Страница настройки соединения с БД

Наконец, на последнем шаге установки появляется окно входа, в котором введем данные главного администратора системы, указанные в одном из предыдущих шагов установки.



Рисунок 13 – Страница авторизации

Наконец, мы получаем доступ в CRM-систему. На этом разворачивание инфраструктуры окончено. Для каждого сотрудника организации, которому потребуется доступ к CRM-системе, потребуется создать два аккаунта – в сервере OpenVPN Access Server для установления VPN-соединения, а также непосредственно в SugarCRM.

Заключение

Использование CRM-систем и информационной системы SugarCRM в частности может стать одним вариантом для совершенствования функциональных возможностей существующих в РФ информационных систем, эксплуатируемых в управляющих организациях (товариществах собственников жилья). Также, внедрение CRM-систем (например, SugarCRM) может стать одним из шагов для создания единого информационного пространства, включающего собственников жилья в контур управления многоквартирными домами. При этом SugarCRM может быть выбрана для взаимодействия с клиентами (собственниками жилья) как информационная система, не предъявляющая к организации серьезных требований к уровню готовности к автоматизации, а также к IT-инфраструктуре управляющей организации.

Библиографический список

1. Попов А.А. Разработка облачного информационного сервиса для функционирования инновационной IT-инфраструктуры организации по управлению многоквартирными домами // Известия Российского экономического университета им. Г.В. Плеханова. 2013. №4(14). С.19-57
2. Телемтаев М.М., Попов А.А. Анализ информационных систем, используемых за рубежом организациями по управлению недвижимостью и собственниками (арендаторами) жилья // Известия Российского экономического университета им. Г.В. Плеханова. 2012. №1(6). С.92-163
3. Черников Б.В., Попов А.А. Выбор информационной системы с учетом уровня готовности предприятия к информатизации // Информатизация и связь. 2016. №3. С.152-159
4. Crist E.F., Keijser J.J. Mastering OpenVPN. London: Packt Publishing, 2015. 367 с.
5. Bearnes B.. How To Install Linux, Apache, MySQL, PHP (LAMP) stack on

-
- Ubuntu 16.04. // Digital Ocean. 2016. URL: <https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-16-04>
6. The Apache Software Foundation. Apache HTTP Server Version 2.4 Documentation. // Apache.org. 2016. URL: <https://httpd.apache.org/docs/2.4/>
7. Хокинс С. Администрирование веб-сервера Apache и руководство по электронной коммерции. М.: Вильямс, 2001. 336 с.