

# **SIGVI R2**

Guía de Instalación

# Índice

1. Introducción.....	3
2. Requerimientos.....	4
2.1. Plataforma.....	4
2.2. Servidor Web.....	4
2.3. Servidor de bases de datos relacionales.....	4
2.4. Intérprete de PHP5.....	4
3. Instalación.....	5
3.1. Datos de sistema.....	5
3.2. Instalación de la aplicación.....	6
3.3. Creación de la base de datos.....	6
3.4. Adaptando la aplicación a su entorno.....	6
3.5. Comprobar accesos LDAP.....	7
3.6. Configuración de los procesos batch.....	7
3.7. Accediendo por primera vez.....	8
4. Más ayuda.....	8

# 1. Introducción

SIGVI son las siglas de Sistema Inteligente de Gestión de Vulnerabilidades Informáticas. Es una herramienta para poder gestionar las vulnerabilidades de nuestros sistemas.

Este proyecto se desarrolla y mantiene desde UPCnet, empresa de servicios TIC del grupo UPC (Universidad Politécnica de Cataluña). También ha sido co-financiado durante el 2008 por el Ministerio de Industria, Turismo Y Comercio de España (MITYC, [www.mityc.es](http://www.mityc.es)) para la obtención de un producto precompetitivo.

El SIGVI es una aplicación Web compuesta por un conjunto de scripts programados en PHP que implementan la lógica de la aplicación y una base de datos relacional donde se guardan los datos. Algunos scripts se ejecutan como procesos batch (generalmente por la noche) para realizar las tareas que no requieren la interacción humana, como por ejemplo las cargas de vulnerabilidades desde las fuentes, el chequeo de las vulnerabilidades en nuestros sistemas, etc. El resto son los scripts que programan la propia aplicación Web.

Este documento trata de detallar cómo instalar una nueva instancia del SIGVI.

## 2. Requerimientos

El SIGVI R2 es una aplicación Web implementada en PHP5 y que necesita una base de datos relacional para almacenar los datos.

### 2.1. Plataforma

La aplicación ha sido desarrollada y testada sobre plataformas Linux. El proceso de instalación será el mismo en cualquier plataforma UNIX (salvando los directorios Web, que dependerá más del propio servidor Web). Dado que es un lenguaje interpretado también funcionará en plataformas Windows, aunque hay diferencias en la configuración de los procesos batch.

Se recomienda instalar la aplicación en una plataforma Linux.

### 2.2. Servidor Web

Será necesario un servidor Web que permita interpretar scripts de PHP5. Esta aplicación ha sido desarrollada y testada en un entorno Web basado en **Apache Web Server, versión 2** con el módulo de **PHP5** (opcionalmente con el módulo SSL).

Si su servidor Web es otro diferente, debe tener en cuenta que la aplicación se basa en un sistema de directorios. Existen directorios comunes, donde se alojan las clases, imagenes, iconos, etc. Y otros directorios que contienen la lógica propia de la aplicación.

Las referencias internas requieren tanto accesos relativos y accesos absolutos.

Ejemplo de instalación en sistemas basados en “apt”:

```
> apt-get install apache2-mpm-prefork libapache2-mod-php5
```

### 2.3. Servidor de bases de datos relacionales

SIGVI R2 proporciona una capa de abstracción de bases de datos, permitiéndole funcionar sobre cualquier tipo que implemente esa capa. Esta versión se provee con acceso a RDBMs MySQL, Postgres y Oracle.

Pero esto es únicamente a nivel funcional, dado que el fichero de carga inicial de la base de datos corresponde a MySQL. En futuras versiones está previsto solventar esta parte proveyendo inicialmente el código necesario para el resto de servidores soportados.

Se recomienda tener instalada tener como servidor de bases de datos MySQL Server versión 5.0 o superior. En otro caso será necesario modificar el fichero SQL de carga inicial para adaptarlo a su sistema.

Ejemplo de instalación en sistemas basados en apt:

```
> apt-get install mysql-server-5.0
```

### 2.4. Intérprete de PHP5

Como ya se ha indicado, la aplicación está implementada en PHP5. Cuando se habla sobre los requerimientos del servidor web, ya se ha explicado que deberá ser capaz de interpretar scripts PHP5.

Pero además es necesario disponer de varios módulos de PHP5:

- Librerías de acceso a su servidor de bases de datos (mysql, oracle, postgres)

Son necesarias las librerías de acceso a la base de datos. Si por ejemplo usa MySQL como servidor de base de datos necesitará el módulo de PHP con las librerías necesarias (módulos php5-mysql, php5-pgsql, ...).

- Librería de generación de imágenes GD

La aplicación necesita generar gráficas para los resúmenes y gráficas de estado. Necesita tener instalada el módulo de PHP para generar imágenes (módulo php5-gd).

- Intérprete de línea de comandos

Parte de la aplicación son scripts que forman procesos batch, es decir, se ejecutan en diferido y no se interpretan a través del servidor Web, si no directamente desde el sistema. Para ello es necesario disponer del módulo de PHP de línea de comandos (php5-cli).

### 3. Instalación

El SIGVI R2 se distribuye en un paquete comprimido con los comandos tar y gzip. Éste contiene:

- Un fichero ChangeLog, donde se explican las últimas novedades aportadas en la versión del paquete
- Un fichero INSTALL donde se explica brevemente cómo instalar la aplicación
- Un fichero SQL para crear la base de datos de la aplicación y realizar la carga inicial.
- El árbol de directorios y scripts que forman la propia aplicación.

Primero deberá extraer el contenido del paquete:

```
> tar xzvf sigvi-xxxx.tgz
```

#### 3.1. Datos de sistema

Para explicar la instalación tomaremos como ejemplo un entorno concreto con la siguiente configuración:

Descripción del software:

- Sistema Operativo: Ubuntu Desktop 7.04
- Servidor Web: Apache Web Server 2.2.8-1 con módulo PHP5
- Intérprete de línea de comandos PHP5
- Servidor de bases de datos MySQL 5.0.51a

Descripción del entorno:

- Directorio raíz del servidor Web: /var/www
- Usuario sobre el que se ejecutan los procesos de Apache: www-data
- Usuario de sistema con el que realizaremos la instalación: root
- Usuario de base de datos MySQL con permisos para crear bases de datos: root

Deberá adaptar las referencias a los de su sistema.

### **3.2. Instalación de la aplicación**

Dentro del paquete existe un directorio llamado “sigvi” que contiene la aplicación. Debemos mover ese directorio dentro del árbol de directorios de nuestro servidor Web (en algunas versiones se llama htdocs):

```
# mv sigvi /var/www/sigvi
```

Luego tendremos que modificar los permisos de éste directorio para que el usuario con el que se ejecuta Apache pueda acceder a los scripts a través de los directorios. En las últimas versiones de Apache éste usuario se llama “www-data”, pero puede consultarlo mirando el propietario de los procesos, por ejemplo “apache” que puede visualizar con el comando “ps -efa”

```
# chown -R www-data:www-data /var/www/sigvi
```

```
# chmod -R 750 /var/www/sigvi
```

A partir de este momento el servidor Web ya tendrá acceso a la aplicación, aunque aún no funcionará dado que falta por completar el resto de la instalación.

### **3.3. Creación de la base de datos**

Como ya se ha comentado, SIGVI R2 provee una capa de abstracción de base de datos que le permitiría trabajar con cualquier tipo de RDBMS. No obstante el fichero SQL mediante el cual se genera la base de datos y se inicializa pertenece a MySQL.

Para crear la base de datos necesitaremos usar un usuario de MySQL con privilegios de creación, que en este caso es root.

Para realizar la carga inicial, en la cual ya se crea la base de datos usaremos el fichero .sql que encontraremos dentro del paquete, en el mismo directorio donde está el fichero INSTALL. Este fichero .sql se completa con la versión del SIGVI que estemos instalando. Por ejemplo, si estamos instalando la versión 1.3.0 del SIGVI R2 ejecutaremos:

```
# mysql -u root -p < sigvi-1.3.0.sql
```

Al finalizar, si no se han producido errores, tendremos creada la base de datos del SIGVI, junto con un usuario “sigvi” y todas las tablas inicializadas.

### **3.4. Adaptando la aplicación a su entorno**

Una vez completados los pasos anteriores faltará configurar la aplicación para poder acceder a ella vía Web. Para ello tendremos que editar el fichero app.conf.php que se encuentra dentro del directorio “conf” del SIGVI.

Concretamente tendrá que revisar los parámetros HOME, ADM\_EMAIL, SERVER\_URL, y la configuración de la base de datos (al final del fichero).

#### **3.4.1. HOME y SERVER\_URL**

Tenga especial atención con la constante HOME, dado que es la constante que se usa internamente para

hacer referencia a la propia aplicación a través del navegador. Dependerá de la URL que usemos para acceder a la aplicación, es decir, el path que viene detrás de la URL del propio servidor.

¿Cómo definir correctamente esta constante?

Estos son ejemplos de distintas situaciones de instalación de la aplicación y cómo definirla en cada caso.

- Para acceder al SIGVI usamos una URL como esta: <http://server.localdomain.domain/sigvi>  
Entonces el valor de HOME será: `define("HOME", "/sigvi")` prestando atención a la barra final.  
El valor del SERVER\_URL será: `define("SERVER_URL", "server.localdomain.domain")`
- Usamos una URL como esta: [http://server.localdomain.domain/my\\_applications/sigvi\\_r2](http://server.localdomain.domain/my_applications/sigvi_r2)  
Entonces el valor de HOME será: `define("HOME", "/my_applications/sigvi_r2")`  
El valor del SERVER\_URL será: `define("SERVER_URL", "server.localdomain.domain")`
- Tenemos instalado el SIGVI en un Virtual Host, y para acceder a él usamos esta URL: <http://server.localdomain.domain:81/>  
Entonces el valor de HOME será: `define("HOME", "")`, importante ver que no hay “/”.  
El valor del SERVER\_URL será: `define("SERVER_URL", "server.localdomain.domain:81")`

### **3.5. Comprobar accesos LDAP**

El SIGVI permite validar las contraseñas de los usuarios contra servicios externos. Para ello bastará tener el plugin necesario. En esta versión se provee un plugin para autenticación vía LDAP.

Si en su entorno se realizan las validaciones vía LDAP podrá configurar el SIGVI para que valide los usuarios que cree en él contra su servicio. Para configurarlo vaya al final del fichero `app.conf.php` dentro del directorio `conf` y adáptelo a su entorno.

Los usuarios que tengan el campo “externo” a cierto, se validarán usando este método.

### **3.6. Configuración de los procesos batch**

Los procesos batch, o procesos diferidos son los encargados de actualizar los datos de la base de datos como por ejemplo descargar las vulnerabilidades y actualizar la tabla de vulnerabilidades, comprobar qué productos, de los que tiene instalados en sus servidores, están afectados por alguna vulnerabilidad, avisar a los administradores, etc.

Aunque la aplicación provee de mecanismos a través de la interfaz Web para ejecutar manualmente los procesos más importantes, es aconsejable la configuración de ejecución en diferido para automatizar las tareas y minimizar el seguimiento y el fallo humano.

En los sistemas UNIX la configuración se realiza vía `crontab`, que puede realizarse de dos maneras: usando el fichero de sistema `/etc/crontab` o bien el `crontab` del usuario (comando `crontab -l` para visualizar, o `crontab -e` para editarlo).

Primero tendremos que editar el fichero `cron.sh` que se encuentra dentro del directorio `cron` del SIGVI. Revise las tres primeras variables:

- PHP: Que indica el path al ejecutable del intérprete de php de línea de comandos (generalmente

/usr/bin/php5)

- DIR: el directorio donde ha instalado el SIGVI (en este ejemplo es /var/www/sigvi
- OUTPUT: el fichero de sistema donde se redirigirá la salida estándar y de error de los procesos.

Luego use el método que prefiera, ya sea editando el fichero /etc/crontab o usando el crontab de un usuario (crontab -e). Tenga en cuenta que el usuario que use deberá tener acceso al directorio Web donde está alojado el SIGVI (por ejemplo www-data, o root).

- Opción 1: /etc/crontab. Como root, editaremos el fichero /etc/crontab y agregaremos una línea como ésta:

```
0 5 * * * www-data /var/www/sigvi/cron/cron.sh > /dev/null 2>&1
```

En este caso hemos indicado que el usuario bajo el cual se ejecutará el proceso es www-data, que tiene acceso al directorio Web del SIGVI. También deberá adaptar el directorio indicado (/var/www/sigvi) al de su instalación.

- Opción 2: crontab -e. Ya sea como root, o como un usuario con acceso al directorio Web (por ejemplo www-data) ejecutamos el comando “crontab -e” y agregamos la siguiente línea:

```
0 5 * * * /var/www/sigvi/cron/cron.sh > /dev/null 2>&1
```

Adaptando el directorio al de su instalación.

Finalmente se aconseja ejecutar manualmente el shell script cron.sh y comprobar la salida, para detectar y corregir posibles errores de configuración.

Si su plataforma es Windows, deberá crear una tarea programada para ejecutar los scripts .php tal como hace el shell script cron.sh:

- plugins/sources/load\_vulnerabilities.php
- plugins/sources/check\_server\_vulnerabilities.php
- plugins/discoverer/cron/check\_repository\_updates.php

### **3.7. Accediendo por primera vez**

Abra un navegador Web y acceda al servidor (por ejemplo <http://server.localdomain.domain/sigvi>) y accederá a la página de login.

Por defecto existe un único usuario: “admin”, cuya contraseña es “admin”.

Cambie la contraseña lo antes posible.

Una vez validado accederá al menú principal. En ésta encontrará en la parte superior del menú un icono de información a través del cual podrá acceder a una página donde se le indicará cuáles son los pasos a seguir tras instalar una nueva instancia del SIGVI.

Para más información acerca de las tareas iniciales, lea por favor el manual del usuario, en el apartado de cada perfil encontrará un apartado de tareas iniciales donde se explica con más detalle qué tareas debe ejecutar para dejar lista su nueva instancia.

## **4. Más ayuda**

Si necesita más ayuda puede pedir información a [sebastian.gomez@upcnet.es](mailto:sebastian.gomez@upcnet.es)

Gracias por su interés, esperamos que le resulte útil.