

Apéndice A

Instalación y configuración de Maya

A.1. Requisitos

Para un correcto funcionamiento del sistema se deben tener instalados una serie de componentes tanto en punto de control como en los puntos de acceso. En las siguientes subsecciones se explicará detenidamente la instalación de cada una de ellas.

A.1.1. Punto de control

1. APACHE: versión 2.0 o superior. Además deberá estar configurado para funcionar con OpenSSL. En el punto A.3.1.1 se detalla los pasos a seguir para ello.
2. MySQL: versión 5.0 o superior. Se recomienda así mismo disponer de una herramienta como MySQL Query Browser que facilita la realización de consultas y visualización de tablas de la base de datos.
3. PHP: versión 5.0 o superior. Debe estar configurado correctamente para funcionar con Apache y MySQL. Al igual que en el caso anterior, se explicará paso a paso su instalación.
4. OpenSSL: versión 0.9.8d. Es un requisito necesario por PHP ya que va a emplear funciones de acceso por SSH (libssh2) y necesita disponer de estas librerías.
5. AODV-UU: Modificación del AODV de la Universidad de Uppsala, versión 0.93 o superior. El protocolo de encaminamiento necesario para poder formar parte de la red *ad-hoc*.

6. OpenWRT SDK¹: *Software Development Kit* de OpenWRT[?], nos servirá para desarrollar nuestras propias aplicaciones para OpenWRT. Incluye numerosas utilidades como el compilador cruzado que permitirá compilar código para una máquina (como el punto acceso) que usa una arquitectura diferente al equipo dónde realizamos la compilación.

A.1.2. Puntos de acceso

1. OpenWRT: versión WhiteRussian RC6. La interfaz de la aplicación ha sido desarrollado sobre dicha versión (que es la más reciente a la finalización de este proyecto), por tanto se recomienda altamente su uso, ya que la interfaz gráfica varía ligeramente de una a otra.
2. AODV-UU: El protocolo de encaminamiento necesario para poder formar parte de la red *ad-hoc*, este será un componente que funcionará tanto en el punto de control como en los puntos de acceso de la red.
3. Libgcc: paquete disponible para instalación con el OpenWRT. Esta librería es necesaria, ya que vamos a utilizar código compilado en C en los puntos de acceso.
4. Libopenssl²: librería necesaria para el sistema de cifrado de clave pública por el método RSA.

A.2. Copia e instalación de los ficheros del programa

Suponiendo que tenemos todos los componentes que hemos enumerado ya se encuentren correctamente instalados en nuestro sistema (en caso de no ser así, en el punto A.3 se explica detenidamente la instalación de todos ellos), vamos a detallar los pasos a seguir para la instalación y configuración de nuestra aplicación.

Debemos señalar que toda la configuración de los puntos de acceso que vamos a explicar en el punto A.2.1, están preparados para funcionar sobre el modelo *Linksys WRT54G*. Esto implica que los procesos están ya compilados para arquitectura MIPS, la interfaz inalámbrica es la *eth1* y otras características, por tanto, recomendamos al usuario que vaya a realizar la instalación en otro dispositivo leer el punto A.2.3, dónde se detallará los aspectos que debe tener en cuenta.

¹ Este no es un requisito indispensable, ya que en este proyecto facilitamos la versión compilada del protocolo tanto para i386 como para MIPS, pero en caso de que el usuario desee compilar otra versión, otro protocolo, o cualquier otra utilidad siempre le será de gran ayuda este *software* de desarrollo.

² Actualmente la versión que proporciona OpenWRT de esta librería es la 0.9.8d (la más reciente), es muy posible que necesitemos la misma versión de OpenSSL en el equipo que realiza las funciones de punto de control.

A.2.1. Puntos de acceso

1. Debemos copiar la carpeta *MAYA* a cada uno de los puntos de acceso de la red *mesh*. Es importante que esta carpeta se encuentre en el directorio raíz de OpenWRT. Para ello podemos utilizar SSH y realizar la copia mediante el comando: "`scp -r path_a_la_carpeta_MAYA ip_del_dispositivo`".
2. A continuación debemos acceder al sistema operativo del router mediante SSH, ejecutando simplemente: "`ssh ip_del_dispositivo`"³. Llegado a este punto, si es la primera vez que accedemos al punto de acceso después de instalar OpenWRT, deberemos configurar el password vía *web* en la página de configuración del *router* (`http://ip_del_dispositivo`), donde se nos informará que es la primera vez que accedemos al sistema y debemos introducir una clave nueva. Una vez realizado este paso, no deberíamos tener problema para acceder por SSH.
3. El siguiente paso es acceder a la subcarpeta *init_scripts*, donde ejecutaremos el *script install.sh* de la forma: "`sh install.sh`". La función de este pequeño *script* es configurar el punto de acceso para ejecutar varios procesos cada vez que arranque. Estos procesos son: arrancar el servidor UDP para la recepción de mensajes, así como el protocolo de encaminamiento, y, en caso de que sea la primera que se inicia el dispositivo, generar su clave pública y privada. Para ello añade en el directorio `/etc/init.d` (donde se encuentran los procesos que se ejecutarán al arrancar el sistema) enlaces simbólicos a los *scripts* que realizan las funciones anteriormente comentadas. Después de realizar esta operación el *router* se reiniciará automáticamente y la instalación estará completada.

A.2.2. Punto de control

1. Para empezar, debemos copiar los archivos que se encuentran en la carpeta *Maya_web* en el directorio donde tengamos configurado nuestro servidor *Apache* para mostrar las páginas. Para que no haya problemas a la hora de generar los archivos necesarios para el correcto funcionamiento de la aplicación, debemos asegurarnos que dicho directorio cuenta con todos los permisos de lectura escritura, al menos para el usuario que crea *Apache* (*daemon*) que será el que ejecute los comandos de nuestra aplicación.
2. Una vez hayamos otorgado los permisos, deberemos acceder a la carpeta *Maya_web*, ya correctamente ubicada, y accederemos al fichero *params.inc*, donde deberemos configurar algunos parámetros:
 - `w_interface`: el nombre de nuestra interfaz inalámbrica (por defecto `wlan0`).

³OpenWRT, sólo cuenta con el usuario *root*, por tanto debemos acceder desde nuestro sistema siendo superusuario.

- `default_password`: el *password* por defecto que utilizará la aplicación para acceder vía SSH a los dispositivos.

A continuación, en el fichero *dbopen.inc*, se establecen el nombre y la contraseña para el acceso a la base de datos, debemos asegurarnos que usaremos en el punto 4 los mismos valores que aquí indiquemos, para que se pueda producir la conexión con la base de datos.

Por último ejecutaremos el *script* *rsa.sh* que se halla en la misma carpeta ejecutando para ello *"sh rsa.sh"*. Esto nos permitirá generar las claves pública y privada y el HIT del punto de control.

3. Así pues, también deberemos dar privilegios al usuario *daemon* para que pueda ejecutar comandos de configuración de nuestra tarjeta *wireless*. Para ello deberemos debemos añadir la línea siguiente en el archivo de sistema *sudoers* (podemos acceder directamente a editar este archivo usando el comando de consola *visudo*) *"daemon ALL=NOPASSWD:/LOCAL/sbin/iwconfig"*.
4. Cómo ultimo paso deberemos cargar el esquema de la base de datos Maya. Para conseguirlo, deberemos generar la base de datos ejecutando *"mysqladmin create maya -p"* y cargaremos el esquema que se proporciona en la carpeta Database: *"mysql maya <maya.sql -p"*. Ahora tan sólo nos resta, crear el usuario Maya, ya que las peticiones a la base de datos se realizarán con este nombre. En primer lugar, entraremos en *mysql* como superusuario ejecutando *"mysql -u root -p"* y a continuación crearemos el usuario y le otorgaremos permisos con la instrucción *"grant all on maya.* to maya@localhost identified by password_del_punto_2"*.

A.2.3. Compilación y adaptación a otras plataformas

En este apartado vamos a explicar las características a tener en cuenta cuando queramos compilar los ficheros de la aplicación a otra plataforma.

En la carpeta *source*, encontraremos los ficheros fuente de los procesos que utiliza nuestra aplicación, por un lado los ficheros de los *sockets udp (client, server y clientCP)* y por otro el protocolo de encaminamiento, *AODV-UU* (que se explicará en puntos posteriores). Además, para cada uno de ellos ya encontramos versiones compiladas para MIPS y X86. No obstante, es posible que el usuario quiera recompilarlos, bien para esa plataforma o bien para otra, especialmente si se está usando un modelo de punto de acceso diferente.

Para realizar la compilación a otra plataforma es requisito indispensable disponer de un compilador para esa arquitectura objetivo y, cómo se verá más adelante, disponer de las librerías OpenSSL, ya que se hacen uso de funciones que se encuentran definidas en ellas. También será necesario acceder al fichero *sockets.h*, dónde deberemos indicar cuál es la interfaz inalámbrica en nuestro dispositivo en la función *get_wlan_ip* antes de realizar la compilación.

Con todo ello, ya estaría todo a punto para realizar la compilación ejecutando: *"compilador_cruzado_C -lssl archivo_fuente -o archivo_destino"*. Recorde-

mos que la opción `-lssl` nos permitirá utilizar todas las funciones incluidas relacionadas con el sistema de claves.

A.3. Instalación de los programas necesarios

A.3.1. Punto de control

A.3.1.1. Apache

A continuación vamos a mostrar los pasos básicos para la instalación del servidor Apache en nuestro computador. No se aconseja el uso de paquetes que proporcionan las distribuciones ya que no pueden ser configurados con las opciones necesarias antes de ser instalado.

1. En primer lugar nos dirigiremos a la página oficial de esta aplicación <http://httpd.apache.org/> y descargaremos la versión que deseemos. El sistema Maya ha sido probado con la versión 2.2.2 de Apache.
2. Una vez se ha desargado el archivo, accederemos al directorio donde lo hemos descargado y lo descomprimiremos usando: `"tar xvf httpd-2_0_NN.tar.gz"`.
3. Accederemos a la carpeta generada: `"cd httpd-2_0_NN.tar.gz"`, y configuraremos Apache con las siguientes parámetros: `"./configure -prefix=/Apache -enable-so -enable-deflate -enable-ssl"`.
4. Por último ejecutar en línea de comandos: `"make && make install"` para completar la instalación del servidor.

Si todo ha funcionado correctamente el servidor debe estar listo para funcionar en nuestro computador. Sin embargo, antes de ponerlo en funcionamiento deberemos introducir unos cambios necesarios en su archivo de configuración que se encuentra en `"directorio_apache/conf/httpd.conf"`, a continuación los detallamos:

1. En la sección *Dynamic Shared Object Support* debemos añadir la línea `"LoadModule php5_module modules/libphp5.so"`, esto nos le permitirá reconocer el código PHP, algo esencial en nuestra aplicación.
2. Buscar la cadena *Addtype* y añadir al final de los tipos insertados que le siguen, los documentos PHP de la forma: `"AddType application/s-httpd-php .php .phtml"` y `"Addtype application/x-httpd-php-source .phps"`⁴. Esto permitirá a Apache reconocer los documentos a partir de la extensión del archivo.

⁴Este proyecto no cuenta con archivos `.phtml` ni `.phps`, pero es posible que sean usados en posibles ampliaciones, por lo que puede ser cómodo tener escrita por adelantado esta configuración.

3. Buscar el la cadena *Documentroot* y colocar aquí el directorio donde vayamos a guardar los ficheros de nuestra página web. A continuación habra que modificar más abajo la línea "*<Directory "/www/">*" sustituyendo *www* por el directorio que hayamos indicado en *Documentroot*.
4. La página *index.php* se encuentra entre las páginas por defecto que carga el servidor, por tanto debemos añadirlas. Para ello, localizaremos la cadena *DirectoryIndex* de forma que queda igual que: "*DirectoryIndex index.html index.html.var index.php index.htm*". Hay que destacar que el orden influye en el orden de prioridad del servidor a la hora de qué páginas buscar.
5. Asegurarse que el usuario del servidor, es el usuario apache (*daemon*). Si este usuario no existe en el sistema, añadirlo a los usuarios Linux.
6. Ya podemos arrancar nuestro servidor con el comando: "*directorio_apache/bin/apachectl*".

A.3.1.2. MySQL

Para instalar MySQL, deberemos dirigirnos a su página oficial <http://www.mysql.com/> y dscargar los paquetes de servidor y cliente correspondientes para nuestra distribución de Linux. Los pasos a seguir en la instalación dependen de la distribución por tanto aconsejamos seguir las instrucciones de instalación que se ofrecen en la propia página. También aconsejamos la descarga de dos aplicaciones que no facilitarán la tarea a la hora de administrar nuestra base de datos que son: MySQL Query Browser y MySQL Administrator.

Una vez instalado el servidor, conviene añadir un password al administrador por razones de seguridad. Para ello, como *root* escribir: "*mysqladmin password cualquier_password*". Ahora lo que debemos hacer es introducir los esquemas en la base de datos que usará el sistema Maya.

A.3.1.3. PHP

Para la instalación de PHP y su compatibilidad con Apache y MySQL debemos seguir los siguientes pasos:

1. Accederemos vía web a la página <http://www.php.net/downloads.php> donde descargaremos la última versión de PHP. El sistema Maya ha sido probado con la versión 5.2.0.
2. Una vez se ha descargado el archivo, accederemos al directorio donde lo hemos descargado y lo descomprimiremos usando: "*tar xzvf php-5.N.N.tar.gz*".
3. Accederemos a la carpeta generada: "*cd php-5.N.N.tar.gz*", y configuraremos Apache con las siguientes parámetros: "*./configure --with-apxs2=/Apache/bin/apxs --with-mysql*".
4. Una vez configurado, procederemos a la instalación, usando: "*make & make install*".

5. A continuación debemos configurar el fichero *php.ini*, lo más cómodo es trabajar con el archivo de ejemplo que trae el propio paquete, para esto: `"cp php.ini-dist /usr/local/lib/php.ini"`.
6. Deberemos cambiar el parámetro *doc_root* que se halla en el archivo de configuración, y escribir la ruta en la cual se guardan los sitios web a mostrar.

A.3.1.4. OpenSSL y Libssh2

Vamos a explicar el procedimiento a seguir para instalar estas librerías, necesarias para el uso de funciones SSH desde código PHP y para generar las claves pública y privada. OpenSSL suele estar disponible para todas las distribuciones de Linux como paquete instalable por el gestor de software de la distribución, sin embargo, explicaremos paso a paso como descargarlo desde la *web* ya que servirá para todos los usuarios.

1. Accederemos a página <http://www.openssl.org> de dónde descargaremos la versión correspondiente (la misma que dispongamos para descargar en el gestor de paquetes de OpenWRT, ello lo podemos comprobar en el punto A.3.2.3) en la sección *source*.
2. Descomprimiremos el fichero accediendo al directorio donde se encuentra y ejecutando `"tar zxvf openssl-X.X.X.tar.gz"`.
3. Por último, situados dentro de la carpeta que hemos extraído teclearemos `"./configure && make && make install"`.
4. Repetiremos exactamente los 3 primeros pasos, para el archivo *libssh2-X.X.tar.gz* que lo podemos obtener en la <http://sourceforge.net/projects/libssh2/>.
5. A continuación ejecutaremos el instalador *pear* para PCL/ssh2: `"pear install ssh2"`, esto nos generará el archivo *ssh2.so*.
6. Copiaremos el archivo anterior al directorio especificado en nuestro fichero *php.ini* detrás de la etiqueta *extensions_dir*.
7. Por último añadiremos al nuestro archivo *php.ini* la línea *extension=ssh2.o* detrás de la etiqueta *dynamic extensions* del fichero, guardaremos los cambios y reiniciaremos el servidor web (en nuestro caso *Apache*) en caso de que lo tengamos funcionando.

A.3.1.5. OpenWRT SDK

Recomendamos la instalación de esta utilidad para todo tipo de usuario ya que a parte de proporcionar muchas ventajas a los más expertos como la creación de sus propios paquetes de distribución o la creación de su propia versión de OpenWRT, tiene herramientas básicas como el compilador cruzado que podemos necesitar para recompilar código del programa a los que hemos modificado los parámetros. Los pasos a seguir son:

1. Descargar de <http://downloads.openwrt.org/whiterussian/newest/> el archivo *OpenWrt-SDK-Linux-i686-1.tar.bz2*.
2. Para finalizar, descomprimiremos el fichero (no hay que instalar nada) con el comando `"tar xvf OpenWrt-SDK-Linux-i686-1.tar.bz2"`.

A.3.1.6. AODV-UU

A continuación se detallan los pasos para la instalación del protocolo AODV-UU en nuestro ordenador:

1. Nos dirigimos a la página <http://core.it.uu.se/core/index.php/AODV-UU> dónde descargaremos la versión de AODV-UU más reciente. Maya ha sido probado con la versión 0.9.3.
2. Ejecutaremos el comando `"apt-get install kernel-source-2.6.8 kernel-headers-2.6-386"`, esto nos permitirá instalar las cabeceras y fuentes del *kernel 2.6* de Linux .
3. A continuación, descomprimiremos el archivo que acabamos descargar, para ello nos ubicaremos en el directorio dónde hemos guardado el fichero y ejecutaremos: `"tar xvf aodv-uu-X.X.X.tar.gz"`.
4. A continuación procederemos a la compilación del protocolo de encaminamiento para ello accederemos a la carpeta que se ha creado al descomprimir el archivo: `"cd aodv-uu-X.X.X"` y ejecutaremos el comando `"make && make install"`
5. Como resultado de la compilación obtendremos un archivo llamado *aodvd* y un módulo llamado *kaodv.o* o *kaodv.ko* dependiendo de la versión del kernel. El primero es el archivo ejecutable y se encargará de cargar el módulo en memoria. Por tanto para ejecutar el protocolo tan solo tenemos que ejecutar el comando `"aodvd"`, automáticamente lo asociará a la interfaz dónde se encuentra nuestra tarjeta de red inalámbrica. En caso de que no asocie el protocolo a la interfaz que deseemos, podemos especificarse-lo usando `"aodvd -i nombre_interfaz"`. Podemos también especificar otro tipo de parámetros, ejecutando `"aodvd -help"` la aplicación nos mostrará todas las opciones configurables.

A.3.2. Puntos de acceso

A.3.2.1. OpenWRT

Es altamente recomendado que instalemos la versión WhiteRussian RC6 de OpenWRT, ya que otra versión podría no ser totalmente compatible con nuestra aplicación. Para ello:

1. Accedemos a la *web* <http://downloads.openwrt.org/whiterussian/rc6/bin/> y descargamos el fichero binario *openwrt-wrt54g-squashfs.bin* que como su

nombre indica es para el router empleado por nosotros. En la misma página hay versiones binarias para otros modelos de *router*, por tanto elegiremos el que incluya el nombre de nuestro modelo.

2. A continuación debemos acceder via web a la página del *router* donde deberemos dirigirnos al apartado de "*system upgrade*" (el nombre puede cambiar dependiendo del *firmware* del punto de acceso) y escribir la ruta hasta el archivo binario que acabamos de descargar.
3. Por último para configurar los dispositivos con el objetivo de que trabajen en modo *ad-hoc* dentro de la misma red. Para ello, debemos dirigirnos a la sección *Network* dentro de la página web de configuración e indicar una dirección IP para el dispositivo ya que por defecto aparece la dirección 192.168.1.1 y si no la cambiamos nos encontraremos con más de un dispositivo con la misma dirección. También habrá que acceder a la sección *Wireless* y asegurarnos que está funcionando en modo *ad-hoc* así como asignarle el ESSID y el canal que deseemos para configurar nuestra red.

Este es el procedimiento más sencillo, aconsejado para la mayoría de usuarios. Para usuarios expertos que compilan su propio firmware con en el entorno de OpenWRT se recomienda el uso de un cliente TFTP para su instalación, en la dirección <http://wiki.openwrt.org/OpenWrtDocs/Installing> se explica el procedimiento a seguir.

A.3.2.2. AODV-UU

Si hemos copiado los archivos necesarios al punto de acceso como se indica en el punto A.2.1, contamos con una versión compilada del protocolo AODV-UU v0.9.3 para arquitectura MIPS. Por tanto, solo realizaremos la instalación que aquí se detalla en caso de que queramos usar otra versión de este protocolo o para otra arquitectura.

Para instalar el protocolo AODV-UU en los dispositivos debemos realizar los siguientes pasos:

1. El primer paso es averiguar la arquitectura de nuestro procesador en el dispositivo, muy probablemente sea MIPS (como en nuestro caso el Linksys WRT54G) por lo que necesitaremos hacer uso del compilador cruzado. Para ello debemos tener la ruta al directorio donde se encuentra el compilador cruzado en la variable PATH de nuestro sistema. El ejecutable del compilador se halla en la carpeta *staging_dir_mipsel* dentro de la carpeta de OpenWRT SDK que hemos generado en el punto A.3.1.5. Para añadirlo al PATH ejecutaremos "*PATH=\$PATH:/camino_hasta_SDK/OpenWrt-SDK-Linux-i686-1/staging_dir_mipsel*".
2. Realizaremos la compilación para mips, accediendo a la carpeta de AODV-UU (punto A.3.1.6) y tecleando: "*make mips*".
3. Ya hemos realizado la compilación, sólo nos falta copiar los archivos al dispositivo, para ello ejecutaremos el comando: "*scp aodvd kaodv.* ip.del.dispositivo.destino:/MAYA/aodv*".

4. Reiniciaremos el dispositivo, usando el comando `"reboot"`.

Si hemos ejecutado el script `install.sh` (indicado en el punto A.3.1.6) y hemos reiniciado el dispositivo, el protocolo se cargará en memoria nada más arranque éste, por tanto, al ejecutar el comando `scp` el sistema nos indicará que dichos archivos se encuentran en uso. La solución será acceder al *router* via SSH: `"ssh ip.del.dispositivo.destino"`, ejecutar el comando `"ps -e"` obteniendo como resultado la lista de procesos activos y localizar la ID del proceso `aodvd`. Por último, teclearemos la instrucción `"kill numero_de_proceso_de_aodvd"` y repetiremos el paso 3 y 4.

A.3.2.3. Libgcc y Libopenssl

Para la instalación de estos paquetes, es necesario que el *router* tenga acceso a internet. Para ello, es necesario que un dispositivo proporcione conectividad entre la red inalámbrica *ad-hoc* e internet de forma que pueda acceder a la información acerca de los paquetes actualmente disponibles.

Una vez nos hemos asegurado de esto, accederemos a la interfaz de configuración web de nuestro punto de acceso de la forma `http://ip.del.dispositivo`. accederemos a la sección *system* y dentro de ella a *installed software*. Acto seguido, pulsaremos en la opción *update packages lists* esto nos actualizará todos los paquetes disponibles para OpenWRT. Deberemos instalar los paquetes `libgcc`, `libopenssl` y `openssl-utils`. Para realizar esto, habrá que localizarlos en la lista y pulsar su correspondiente botón *install*.