

iRedMail

- [Actualizar iRedMail](#)
- [Agregar Nodo Cluster iRedMail](#)
- [Comandos SQL iRedMail](#)
- [DKIM en iRedMail](#)
- [Estructura Tablas Sogo](#)
- [Instalar iRedMail](#)
- [Instalar Iredmail Cluster](#)
- [iRedMail moderar listas](#)

Actualizar iRedMail

Pasos previos

Backups

Realizar backup si es posible de la máquina entera (en el caso de máquina virtual).

Realizar backup de las Bases de datos de iRedMail como:

amavisd

iredadmin

iredapd

roundcubemail

sogo

vmail

Realizar backup de la propias bases de datos de Mysql (permisos y usuarios)

Actualización

El proceso de actualización de iRedMail se ejecuta siguiendo los siguientes pasos:

Versión de iRedMail

Actualizar el fichero `/etc/iredmail-release` con el nuevo número de versión de iRedMail

```
nano /etc/iredmail-release
```

iRedMail almacena la versión en el fichero `/etc/iredmail-release` después de la instalación. Esto nos permitirá saber la versión de iRedMail que estamos ejecutando. Por ejemplo

```
root@mx01: ~#cat /etc/iredmail-release  
1.0.1
```

iRedMail no dispone como tal en muchos casos de una versión, sino que se trata de múltiples productos gestionados por un paquete de instalación. Este número nos permite conocer la versión del instalador (y por lo tanto de los módulos instalados)

Actualizar iRedAPD

Actualizamos iRedAPD (servidor de políticas Postfix) a la última versión estable. Para ello ejecutaremos los siguientes procesos. Es importante conocer la versión de iRedAPD correspondiente con el la base sobre la que estamos instalando

```
cd /root
wget https://dl.iredmail.org/yum/misc/iRedAPD-X.Y.tar.bz2
tar xjf iRedAPD-X.Y.tar.bz2
cd iRedAPD-X.Y/tools
bash upgrade_iredapd.sh
service iredapd restart
```

Actualizar iRedAdmin

Actualizamos iRedAdmin (la versión de código abierto) a la última versión estable

```
cd /root
wget https://dl.iredmail.org/yum/misc/iRedAdmin-A.B.tar.bz2
tar xjf iRedAdmin-A.B.tar.bz2
cd iRedAdmin-A.B/tools
bash upgrade_iredadmin.sh
```

Actualizar iRedAdmin Pro

Actualizamos iRedAdmin Pro (la versión de pago) a la última versión estable

```
cd /root
tar xjf iRedAdmin-Pro-SQL-X.Y.Z.tar.bz2
cd iRedAdmin-Pro-SQL-X.Y.Z/
cd tools
bash upgrade_iredadmin.sh
```

Actualización de otros

Comprobar en la documentación si hay que realizar la actualización de otros subsistemas como por ejemplo roundcube, sogo, netdata, amavisd, mlmmjadmin, etc

Script de actualización

Vamos a ver un pequeño script de automatización de las tareas comentadas

#Procedure for upgrading iRedMail

```
cd /root
wget https://dl.iredmail.org/yum/misc/iRedAPD-X.X.tar.bz2
tar xjf iRedAPD-X.X.tar.bz2
cd iRedAPD-X.X/tools
bash upgrade_iredapd.sh
service iredapd restart
cd /root
wget https://dl.iredmail.org/yum/misc/iRedAdmin-X.X.tar.bz2
tar xjf iRedAdmin-X.X.tar.bz2
cd iRedAdmin-X.X/tools
bash upgrade_iredadmin.sh
cd /root
tar xjf iRedAdmin-Pro-SQL-X.X.x.tar.bz2
cd iRedAdmin-Pro-SQL-X.X.X/
cd tools
bash upgrade_iredadmin.sh
cd /root
wget https://dl.iredmail.org/yum/misc/roundcubemail-X.X.C-complete.tar.gz
tar xf roundcubemail-1.4.2-complete.tar
cd roundcubemail-1.4.2/bin
./installto.sh /opt/www/roundcubemail
```

Agregar Nodo Cluster iRedMail

Para agregar un nodo a una instalación existente de iRedMail, deberemos de seguir los siguientes pasos

Instalar Debian

[Configurar el cliente de NTP](#)

Comprobar el fichero de hosts, que contiene la información correcta del nombre del host

instalamos la herramienta de descompresión:

```
apt-get install bzip2
```

descargamos la version de iRedMail a instalar

en este caso

iRedMail-1.3.2.tar.gz

descomprimos el tar.gz

```
tar -xvf iRedMail-1.3.2.tar.gz
```

Se instala cliente del almacenamiento compartido (glusterfs, ceph o NFS)

se configura /etc/fstab

Por ejemplo:

```
gluster01: /gv0 /mnt/mailstore glusterfs defaults, _netdev 0 0
```

Reiniciamos o hacemos un mount -a se copia el fichero config de otro de los nodos (estará en root/iredmail0.0...) al nuevo, al directorio de instalación

```
se ejecuta lo siguiente:
USE_EXISTING_MYSQL='YES' \
  MYSQL_SERVER_ADDRESS='IP_DEL_GALERA' \
  MYSQL_SERVER_PORT='3306' \
```

```
MYSQL_ROOT_USER='root' \  
MYSQL_ROOT_PASSWORD='MIPASSWORD' \  
MYSQL_GRANT_HOST='%' \  
INITIALIZE_SQL_DATA=NO \  
bash iRedMail.sh
```

Se lanza la instalacion

aplica permisos , esto tarda (hacer un mod para eliminar el cambio de permisos)

Luego pregunta , respondemos N

termina y reboot

CERTIFICADOS

el archivo de configuracion esta en:

```
nano /etc/nginx/templates/ssl.tpl
```

COPIAMOS en /etc/ la carpeta cert que contiene una carpeta /micertificado , ahi estaran nuestro pem y nuestro key

en /etc/ssl/private renombramos iredmail.key y subimos nuestro .key para renombrarlo como iRedMail.key

en /etc/ssl/certs renombramos iredmail.crt y subimos nuestro .pem para renombrarlo como iRedMail.crt

Hacemos un ln

```
ln -s /etc/ssl/certs/micertificado.crt /etc/ssl/certs/iRedMail.crt  
ln -s /etc/ssl/private/micertificado.key /etc/ssl/private/iRedMail.key
```

REDIRECCION DEL WEBMAIL POR DEFECTO

ponemos la redireccion hacia sogo por defecto, en:

```
nano /var/www/html/index.html
```

sustituimos mail por sogo

reiniciamos el servicio nginx

INSTALAR IREDADMIN

Copiamos el tar iRedAPD-4.6.tar.gz a /root

Descomprimos

```
tar -czvf iRedAPD-4.6.tar.gz
```

Ejecutamos

```
cd /root/  
tar xjf iRedAdmin-Pro-x.y.z.tar.bz2  
cd iRedAdmin-Pro-x.y.z/tools/  
bash upgrade_iredadmin.sh
```

Repetimos la operación con el IredAPD

descargamos el iRedAPD-4.6.tar.gz

descomprimos

vamos a tools/

```
bash upgrade_iredadmin.sh
```

MAQUETAR SOGO

```
nano /etc/sogo/sogo.conf
```

Modificamos para que coincida con el de producción. modificamos workers

```
nano /etc/default/sogo
```

PREFORK=500 (o lo que ponga en pro)

CONFIGURAMOS MEMCACHED

```
nano /etc/memcached.conf
```

lo dejamos como en producción

REVISAR POSTFIX

```
nano /etc/postfix/main.cf
```

```
nano /etc/postfix/main.cf
```

cotejar con produccion

APARIENCIA Y LOGO

subimos nuestros ficheros a :

`/usr/lib/GNUstep/SOGo/WebServerResources/css/theme-default.css`

`/usr/lib/GNUstep/SOGo/WebServerResources/img/sogo-full.svg`

Comandos SQL iRedMail

Actualizar todos las cuotas de los buzones

```
use vmail;
```

```
update mailbox set quota =25620;
```

También se puede hacer por dominio

```
update mailbox set quota =25620 where username like '%@midominio%';
```

Actualizar la zona horaria y el idioma

```
update mailbox set settings =' timezone: Europe/Madrid;';
```

```
update mailbox set language =' es_ES' ;
```

Deshabilitar sogo para un dominio

Existen 4 campos nuevos en la tabla vmail.mailbox, denominados

enablesogo

enablesogowebmail

enablesogocalendar

enablesogoactivesync

```
update mailbox set enablesogo = 0 where username like '%@midominio%';
```

```
update mailbox set enablesogowebmail = 'n' where username like '%@midominio%';
```

```
update mailbox set enablesogocalendar = 'n' where username like '%@midominio%';
```

```
update mailbox set enablesogoactivesync = 'n' where username like '%@midominio%';
```

DKIM en iRedMail

Generar claves DKIM

```
amavisd-new genrsa /var/lib/dkim/dominio.com.pem 2048
```

```
chown amavis:amavis /var/lib/dkim/dominio.com.pem
```

```
chmod 0400 /var/lib/dkim/dominio.com.pem
```

Agregar claves DKIM al fichero de configuración de amavis

```
nano /etc/amavis/conf.d/50-user
```

Buscar:

```
# Add dkim_key here.
dkim_key('dominio1.es', 'dkim', '/var/lib/dkim/dominio1.es.pem');
dkim_key('dominio2.com', 'dkim', '/var/lib/dkim/dominio2.pem');
```

Un poco más adelante en:

```
@dkim_signature_options_bysender_maps = ( {
    # 'd' defaults to a domain of an author/sender address,
    # 's' defaults to whatever selector is offered by a matching key

    # Per-domain dkim key
    #"domain.com" => { d => "domain.com", a => 'rsa-sha256', ttl => 10*24*3600 },
```

Agregamos (aunque esto ya estará)

```
# catch-all (one dkim key for all domains)
'.' => {d => 'dominio1.es',
      a => 'rsa-sha256',
      c => 'relaxed/simple',
      ttl => 30*24*3600 },
```

```
"dominio2.com" => { d=> 'dominio2.com', a => 'rsa-sha256', c => 'relaxed/simple', ttl => 30*24*3600 },
```

Debemos comprobar que termina en

```
});
```

Comprobar que la configuración es correcta

Para comprobar ejecutamos

```
amavisd-new showkeys
```

Nos aparecerá lo siguiente

```
; key#1 2048 bits, i=dkim, d=dominio1.es, /var/lib/dkim/dominio1.es.pem
dkim._dominio1.mail365.es. [B600 TXT (
  "v=DKIM1; p="
  "MIIBIjANBgqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEAsKiLwJouZh6vqHn1QYWg"
  "XXXXXX".....
  "7wIDAQAB");
```

```
key#2 2048 bits, i=dkim, d=dominio2.com, /var/lib/dkim/dominio2.com.pem
dkim._domainkey.dominio2.com. [B600 TXT (
  "v=DKIM1; p="
  "MIIBIjANBgqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA03gU5SjyduyGFPILfTQh"
  " "XXXXXX".....
  "lwIDAQAB")
```

Agregar los registros DNS

A continuación iremos a nuestra configuración dns y agregaremos el registro (en este caso dkim._domainkey.dominio2.com)

El registro contendrá:

```
"v=DKIM1; p=MIIBIjANBgqhkiG9w0BAQEFAA0CAQ8AMIIBCgKCAQEA03gU5SjyduyGFPILfTQhXXXXXX.....lwIDAC
```

Es decir la palabra clave v=DKIM1; p= y la clave sin retornos de carro ni comillas

Testar los registros DNS

```
root@mail:/# amavisd-new testkeys
TESTING#1 dominio1.es: dkim._domainkey.dominio1.es => pass
TESTING#2 dominio2.com: dkim._domainkey.dominio2.com => invalid (public key: not available)
```

En este caso nos da error puesto que todavía no se han replicado los registros DNS

Estructura Tablas Sogo

La estructura de las tablas en SoGo es la siguiente:

```
-----
-- Fully-Unicode compliant SGO MySQL schema
--
-- Required MySQL version: >= 5.5.5
-- Required SGO version: >= 4.0.0
--
-- This schema includes some specific table creation
-- parameters needed for complete Unicode coverage
-- (required for emoji character support, for instance).
--
-- Note: SGO automatically creates those tables on start,
-- but *without* full Unicode compliance. So this schema
-- must be imported before SGO runs for the first time.
--
-- Strongly suggested MySQL configuration settings
-- (innodb* parameters are mandatory):
--
-- [client]
-- default-character-set          = utf8mb4
--
-- [mysql]
-- default-character-set          = utf8mb4
--
-- [mysqld]
-- character-set-client-handshake = FALSE
-- character-set-server           = utf8mb4
-- collation-server               = utf8mb4_unicode_ci
-- innodb_file_per_table          = TRUE
-- innodb_file_format             = barracuda
-- innodb_large_prefix            = TRUE
--
-- This schema assumes the SGO table names are configured
-- like the following; adjust if needed:
--
-- OCSAclURL                      -> sogo_acl
-- OCSCacheFolderURL              -> sogo_cache_folder
-- OCSEmailAlarmsFolderURL        -> sogo_alarms_folder
-- OCSFolderInfoURL               -> sogo_folder_info
-- OCSSessionsFolderURL           -> sogo_sessions_folder
-- OCSStoreURL                    -> sogo_store
-- SGOUserProfileURL              -> sogo_user_profile
--
-- SGO needs to know MySQL has full Unicode coverage;
-- the following needs to be put in sogo.conf:
```

```
--
-- MySQL4Encoding = "utf8mb4";
--

-----

CREATE TABLE sogo_acl (
[c_folder_id int(11)          NOT NULL,
[c_object      varchar(255) NOT NULL,
[c_uid         varchar(255) NOT NULL,
[c_role        varchar(80)  NOT NULL,
[KEY sogo_acl_c_folder_id_idx (c_folder_id),
[KEY sogo_acl_c_uid_idx (c_uid)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

CREATE TABLE sogo_alarms_folder (
[c_path        varchar(255) NOT NULL,
[c_name        varchar(255) NOT NULL,
[c_uid         varchar(255) NOT NULL,
[c_recurrence_id int(11)      DEFAULT NULL,
[c_alarm_number int(11)      NOT NULL,
[c_alarm_date   int(11)      NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

CREATE TABLE sogo_cache_folder (
[c_uid         varchar(255) NOT NULL,
[c_path        varchar(255) NOT NULL,
[c_parent_path varchar(255) DEFAULT NULL,
[c_type        tinyint(3)   unsigned NOT NULL,
[c_creationdate int(11)     NOT NULL,
[c_lastmodified int(11)     NOT NULL,
[c_version     int(11)      NOT NULL DEFAULT '0',
[c_deleted     tinyint(4)   NOT NULL DEFAULT '0',
[c_content     longtext,
[PRIMARY KEY (c_uid,c_path)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

CREATE TABLE sogo_folder_info (
[c_folder_id   bigint(20)    unsigned NOT NULL AUTO_INCREMENT,
[c_path        varchar(255)  NOT NULL,
[c_path1       varchar(255)  NOT NULL,
[c_path2       varchar(255)  DEFAULT NULL,
[c_path3       varchar(255)  DEFAULT NULL,
[c_path4       varchar(255)  DEFAULT NULL,
[c_foldername  varchar(255)  NOT NULL,
[c_location    varchar(2048) DEFAULT NULL,
[c_quick_location varchar(2048) DEFAULT NULL,
[c_acl_location varchar(2048) DEFAULT NULL,
[c_folder_type varchar(255)  NOT NULL,
[PRIMARY KEY (c_path),
[UNIQUE KEY c_folder_id (c_folder_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

CREATE TABLE sogo_quick_appointment (
[c_folder_id   int(11)       NOT NULL,
```

```

[c_name          varchar(255)  NOT NULL,
[c_uid          varchar(255)  NOT NULL,
[c_startdate    int(11)       DEFAULT NULL,
[c_enddate      int(11)       DEFAULT NULL,
[c_cycleenddate int(11)       DEFAULT NULL,
[c_title        varchar(1000) NOT NULL,
[c_participants text,
[c_isallday     int(11)       DEFAULT NULL,
[c_iscycle      int(11)       DEFAULT NULL,
[c_cycleinfo    text,
[c_classification int(11)     NOT NULL,
[c_isopaque     int(11)       NOT NULL,
[c_status       int(11)       NOT NULL,
[c_priority     int(11)       DEFAULT NULL,
[c_location     varchar(255)  DEFAULT NULL,
[c_orgmail      varchar(255)  DEFAULT NULL,
[c_partmails    text,
[c_partstates   text,
[c_category     varchar(255)  DEFAULT NULL,
[c_sequence     int(11)       DEFAULT NULL,
[c_component    varchar(10)   NOT NULL,
[c_nextalarm    int(11)       DEFAULT NULL,
[c_description  text,
PRIMARY KEY (c_folder_id,c_name)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

```

```

CREATE TABLE sogo_quick_contact (
[c_folder_id    int(11)       NOT NULL,
[c_name         varchar(255)  NOT NULL,
[c_givenname    varchar(255)  DEFAULT NULL,
[c_cn           varchar(255)  DEFAULT NULL,
[c_sn           varchar(255)  DEFAULT NULL,
[c_screename    varchar(255)  DEFAULT NULL,
[c_l            varchar(255)  DEFAULT NULL,
[c_mail         text          DEFAULT NULL,
[c_o            varchar(255)  DEFAULT NULL,
[c_ou           varchar(255)  DEFAULT NULL,
[c_telephonenumber varchar(255) DEFAULT NULL,
[c_categories    varchar(255) DEFAULT NULL,
[c_component     varchar(10)   NOT NULL,
[c_hascertificate int(11)     DEFAULT 0,
PRIMARY KEY (c_folder_id,c_name)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

```

```

CREATE TABLE sogo_sessions_folder (
[c_id          varchar(255) NOT NULL,
[c_value       varchar(255) NOT NULL,
[c_creationdate int(11)     NOT NULL,
[c_lastseen    int(11)     NOT NULL,
PRIMARY KEY (c_id)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;

```

```

CREATE TABLE sogo_store (

```

```
[c_folder_id    int(11)      NOT NULL,  
[c_name        varchar(255) NOT NULL DEFAULT '',  
[c_content     mediumtext   NOT NULL,  
[c_creationdate int(11)      NOT NULL,  
[c_lastmodified int(11)      NOT NULL,  
[c_version     int(11)      NOT NULL,  
[c_deleted     int(11)      DEFAULT NULL,  
[PRIMARY KEY (c_folder_id,c_name)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;
```

```
CREATE TABLE sogo_user_profile (  
[c_uid         varchar(255) NOT NULL,  
[c_defaults    longtext,  
[c_settings    longtext,  
[PRIMARY KEY (c_uid)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 ROW_FORMAT=DYNAMIC;
```

Instalar iRedMail

Instalación de iRedMail

Preparación del entorno

Instalaremos el servidor con una distribución de Linux Debian o RedHat. En nuestro caso usaremos Debian. Si vamos a instalar varios nodos, deberemos de disponer de una base de datos externa y de un almacenamiento compartido

Requisitos

Nombre del servidor

El nombre del servidor se debe de corresponder con el nombre asignado a nuestro servidor de correo. Por ejemplo si vamos a instalar dos o tres servidores MX, y nuestro dominio es dominio.com. Los servidores por ejemplo se podrían llamar:

mx01.dominio.com

mx02.dominio.com

mx03.dominio.com

Para ello en el fichero hosts agregaremos el nombre del servidor editando el fichero `/etc/hosts`

```
nano /etc/hosts
```

Normalmente este fichero contendrá la siguiente información:

```
127.0.0.1    localhost.localdomain    localhost
```

En este caso, suponiendo que la IP de nuestro primer servidor sea la 123.123.123.100 agregaremos esta información al fichero hosts

```
127.0.0.1    localhost.localdomain    localhost
123.123.123.100  mx01.dominio.com    mx01
```

Para comprobar que esta correcto ejecutaremos el comando `hostname -f`


```
root@mx01: ~#hostname -f
mx01.dominio.com
root@mx01: ~#
```

Zona horaria

Para asegurarnos de que el servidor no devuelva correos y no sean devueltos por otros, debemos de configurar la zona horaria y la hora del servidor: ejecutaremos

```
dpkg-reconfigure tzdata
```

Esto nos permitirá configurar nuestra zona horaria correctamente. A continuación instalaremos el cliente ntp

```
apt-get install ntp
```

Y procederemos a configurarlo

```
nano /etc/ntp.conf
```

Aparecerá el fichero de configuración de ntp

```
# /etc/ntp.conf, configuration for ntpd; see ntp.conf(5) for help

driftfile /var/lib/ntp/ntp.drift

# Leap seconds definition provided by tzdata
leapfile /usr/share/zoneinfo/leap-seconds.list

# Enable this if you want statistics to be logged.
#statsdir /var/log/ntpstats/

statistics loopstats peerstats clockstats
filegen loopstats file loopstats type day enable
filegen peerstats file peerstats type day enable
filegen clockstats file clockstats type day enable

# You do need to talk to an NTP server or two (or three).
#server ntp.your-provider.example
server hora.roa.es

# pool.ntp.org maps to about 1000 low-stratum NTP servers.  Your server will
# pick a different set every time it starts up.  Please consider joining the
# pool: <http://www.pool.ntp.org/join.html>
pool 0.debian.pool.ntp.org iburst
pool 1.debian.pool.ntp.org iburst
pool 2.debian.pool.ntp.org iburst
pool 3.debian.pool.ntp.org iburst
```

Después de la línea 18 no habrá nada, por lo que procederemos a añadir en esta línea nuestros servidores ntp que en este caso es el hora.roa.es

Podemos agregar mas servidores de la lista de servidores ntp o bien añadir un pool o modificar el pool por defecto que aparece en las líneas 24 a 27

Instalación en el primer servidor

Instalaremos el bzip2. Nos aseguraremos de estar en la carpeta adecuada por ejemplo root

```
apt-get install bzip2
```

Descargaremos iRedMail

```
wget https://bitbucket.org/zhb/iredmail/downloads/iRedMail-X.Y.Z.tar.bz2
```

Descomprimos

```
tar xjf iRedMail
```

Ahora deberemos de instalar con el siguiente comando, para asegurarnos que la instalación utiliza la Base de Datos externa o el cluster de Base de Datos que hemos instalado

```
USE_EXISTING_MYSQL=' YES' \  
  MYSQL_SERVER_ADDRESS=' 192.168.250.193' \  
  MYSQL_SERVER_PORT=' 3306' \  
  MYSQL_ROOT_USER=' my_user' \  
  MYSQL_ROOT_PASSWD=' MY_PASSWORD' \  
  MYSQL_GRANT_HOST=' %' \  
  bash iRedMail.sh
```

En la instalación nos pedirá la carpeta de almacenamiento para el correo, usaremos la carpeta montada en el almacenamiento compartido que hemos creado anterioremente

Instalación en los demás servidores

Seguiremos los pasos que hemos realizado en el primer servidor, a excepción del comando de instalación, al que tenemos que notificarle que no vuelva a generar los registros de la Base de Datos.

ATENCION a la línea "INITIALIZE_SQL_DATA=NO" es vital para que use la Base de Datos existente y que no dé problemas la instalación

```
USE_EXISTING_MYSQL=' YES' \  
  MYSQL_SERVER_ADDRESS=' 192.168.250.193' \  
  INITIALIZE_SQL_DATA=NO
```

```
MYSQL_SERVER_PORT=' 3306' \  
MYSQL_ROOT_USER=' my_user' \  
MYSQL_ROOT_PASSWD=' MY_PASSWORD' \  
MYSQL_GRANT_HOST=' %' \  
INITIALIZE_SQL_DATA=NO \  
bash iRedMail.sh
```

Instalar iRedMailAPD

Descargamos el iRedMailAPD y ejecutamos

```
cd /root/  
tar zxf iRedAPD-4.6.tar.gz  
cd iRedAPD-4.6.tar.gz/tools  
bash upgrade_iredadmin.sh
```

Instalar iRedAdminPro (Versión de pago)

Descargamos el iRedAdminPro y ejecutamos

```
cd /root/  
tar xjf iRedAdmin-Pro-x.y.z.tar.bz2  
cd iRedAdmin-Pro-x.y.z/tools/  
bash upgrade_iredadmin.sh
```

Instalar Iredmail Cluster

Para Instalar iRedMail, lo primero que deberemos hacer es preparar el servidor

Preparación del servidor

Servicio NTP

Instalaremos y configuraremos el servicio NTP como se explica aquí [Configurar NTP en Debian](#)

Nombre del host

Es imprescindible que el host tenga un nombre canónico que resuelva correctamente para que el servicio de correo funcione. Para comprobarlo, ejecutaremos

```
hostname -f
```

Devolverá un valor que debe ser el nombre de la máquina:

```
mail.ateinco.es
```

Si no devuelve un nombre de dominio completo, deberemos editar el fichero `/etc/hosts`

```
nano /etc/hosts
```

Por ejemplo si tenemos en el archivo `/etc/hosts` lo siguiente:

```
127.0.0.1    localhost.localdomain    localhost
```

Deberemos de modificarlo para que aparezca de la siguiente forma

```
21.85.189.77 mail.ateinco.es    mail
127.0.0.1    localhost.localdomain    localhost
::1         localhost6.localdomain6  localhost6
```

Balanceador para Galera

Necesitaremos instalar un balanceador para que los diferentes nodos de iRedMail, puedan acceder a cualquiera de los servidores de Base de Datos, hay que recordar, que en la configuración, el servidor de Base de Datos a la que apuntará nuestra instalación de iRedMail será la IP del balanceador (Zevenet, HaProxy, Barracuda, etc)

Balanceador para RoundCube, SOGo, iRedAdmin

Necesitaremos instalar un balanceador para que los servicios que dependen de nginx (RoundCube, SOGo, iRedAdmin) o los servicios como IMAP, SMTP o POP3 puedan ser accesibles desde una IP que luego distribuya a cada uno de los nodos

Instalar iRedMail

Instalar con una BBDD Externa

Instalaremos un servidor de [MariaDB](#) o un [Galera Cluster de MariaDB](#)

Inicialización de la BBDD

Crearemos un usuario que se utilizará durante la instalación. Para ello hacemos login en MariaDB

```
root@mail:~# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 1883
Server version: 10.1.44-MariaDB-0ubuntu0.18.04.1 Ubuntu 18.04

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

Y ejecutamos los siguientes comandos

```
GRANT ALL PRIVILEGES ON *.* TO 'admin_iredmail'@%' IDENTIFIED BY 'la_password_que_almacenaremos'
FLUSH PRIVILEGES;
```

```
FLUSH HOSTS;  
set global innodb_force_primary_key = 0;
```

Descarga de iRedMail

Instalamos el bzip2 que necesitaremos para descomprimir el iRedMail

```
apt-get install bzip2
```

Nos movemos a la carpeta root

```
cd /root
```

Descargamos y descomprimos el iRedMail (en el ejemplo usamos la versión 0.9.9, deberemos de escoger la mas moderna.

```
wget https://bitbucket.org/zhb/iredmail/downloads/iRedMail-0.9.9.tar.bz2  
tar xjf iRedMail
```

Instalación en el primer nodo

Vamos a la carpeta del iRedMail, y ejecutamos el siguiente comando

```
USE_EXISTING_MYSQL=' YES' \  
  MYSQL_SERVER_ADDRESS=' 192.168.250.193' \  
  MYSQL_SERVER_PORT=' 3306' \  
  MYSQL_ROOT_USER=' instalar' \  
  MYSQL_ROOT_PASSWD=' la_password_que_almacenaremos' \  
  MYSQL_GRANT_HOST=' %' \  
  bash iRedMail.sh
```

Una vez realizado esto, tendremos el iRedMail instalado en el primer nodo, ahora realizaremos un backup de la configuración de este iRedMail que se encuentra en /root/iRedMail-0.9.9 el archivo se llama config el contenido de dicho archivo será parecido a este

```
export STORAGE_BASE_DIR=' /var/vmail'  
export WEB_SERVER=' NGINX'  
export BACKEND_ORIG=' MARIADB'  
export BACKEND=' MYSQL'  
export VMAIL_DB_BIND_PASSWD=' yLqcXdBy0CLwpX4QTesAjr1J5iq2Rn1'  
export VMAIL_DB_ADMIN_PASSWD=' JF3ffrmR2PbNcu6us2QIIPsBERlV4N1'  
export MLMMJADMIN_API_AUTH_TOKEN=' ImIUhBZT2ao3BFjjadBNjMJry1Nrd01'  
export NETDATA_DB_PASSWD=' RUEUyKajMR7Gak8rPjusj7V416nl5W1'  
export MYSQL_ROOT_PASSWD=' la_password_que_almacenaremos'  
export FIRST_DOMAIN=' ateinco.es'  
export DOMAIN_ADMIN_PASSWD_PLAIN=' la_password_del_postmaster'
```

```
export USE_IREDADMIN=' YES'
export USE_ROUNDcube=' YES'
export USE_SOGO=' YES'
export USE_NETDATA=' YES'
export USE_FAIL2BAN=' YES'
export AMAVISD_DB_PASSWD=' fbYo1C1MREBGaZutqzIzaufFayN2gZ1'
export IREDADMIN_DB_PASSWD=' dTIncixK4vj9f52PzRasiBJEezlv6h1'
export RCM_DB_PASSWD=' 4tnn37hzrsJzLWJSvzx1WCtPaSvlQC1'
export SOGO_DB_PASSWD=' dR7tSorn6saREaEY39Qp3ZQvfc5tvZ1'
export SOGO_SIEVE_MASTER_PASSWD=' owYE3LQdExTjTBGd9gnGjNpwUkj0tK1'
export IREDAPD_DB_PASSWD=' iTGNks97Q2iUg0gRS3NEoMCudvKRCA1'
#EOF
```

Instalar el resto de los nodos

Debemos de ejecutar los mismos pasos: Instalamos el bzip2 que necesitaremos para descomprimir el iRedMail

```
apt-get install bzip2
```

Nos movemos a la carpeta root

```
cd /root
```

Descargamos y descomprimos el iRedMail (en el ejemplo usamos la versión 0.9.9, deberemos de escoger la mas moderna).

```
wget https://bitbucket.org/zhb/iredmail/downloads/iRedMail-0.9.9.tar.bz2
tar xjf iRedMail
```

Y a continuación copiamos el fichero config, que hemos recuperado el primer nodo en la carpeta /root/iRedMail-0.9.9

Ejecutamos el siguiente comando **NOTA** cuidado con la línea INITIALIZE_SQL_DATA=NO ES MUY IMPORTANTE, ya que si no, nos borrará todo lo que había en la base de datos, y fallará la instalación.

```
USE_EXISTING_MYSQL=' YES' \
  MYSQL_SERVER_ADDRESS=' 192.168.250.40' \
  MYSQL_SERVER_PORT=' 3306' \
  MYSQL_ROOT_USER=' instalar' \
  MYSQL_ROOT_PASSWD=' la_password_que_almacenaremos' \
  MYSQL_GRANT_HOST=' %' \
  INITIALIZE_SQL_DATA=NO \
  bash iRedMail.sh
```

Al arrancar nos dirá que iRedMail ya está instalado, que si queremos usar la configuración previamente almacenada, a lo que diremos que sí.

Solución de problemas

Si por lo que sea, hemos cometido algún error, podemos reiniciar la instalación, para lo que previamente habrá que borrar las tablas que se han creado y empezar desde el principio en el primer nodo. Por los ue nos conectaremos al MySQL y ejecutaremos

```
DROP DATABASE amavisd;
DROP DATABASE iredadmin;
DROP DATABASE iredapd;
DROP DATABASE roundcubemail;
DROP DATABASE sogo;
DROP DATABASE vmail;
DROP USER 'amavisd'@'%';
DROP USER 'iredadmin'@'%';
DROP USER 'iredapd'@'%';
DROP USER 'roundcube'@'%';
DROP USER 'sogo'@'%';
DROP USER 'vmail'@'%';
DROP USER 'vmailadmin'@'%';
SELECT host,User FROM mysql.user;
delete from user where user='admin_iredmail' and host='%';
```

Y desde aquí volveremos a crear el usuario, ejecutamos los siguientes comandos

```
GRANT ALL PRIVILEGES ON *.* TO 'admin_iredmail'@'% ' IDENTIFIED BY 'la_password_que_almacenaremos';
FLUSH PRIVILEGES;
FLUSH HOSTS;
set global innodb_force_primary_key = 0;
```

Y podemos empezar de nuevo el proceso.

Otras Configuraciones

Para que iRedMail funcione en este entorno, si vamos a usar un balanceador para los servicios https (iredadmin, roundcube, Solo, etc) deberemos de configurar algunos parámetros como el servicio de memcached, la ubicación de los adjuntos, etc.

<https://docs.iredmail.org/haproxy.keepalived.glusterfs.html>

iRedMail moderar listas

quitar obligacion de moderar una lista concreta:

```
cd /opt/mlmmjadmin/tools
```

```
python3 maillist_admin.py update test@ateinco.com only_moderator_can_post=no
```

```
python3 maillist_admin.py update test@ateinco.com moderated=no
```

quitar obligacion de moderar todas las listas por dominio

```
python3 maillist_admin.py update all@ateinco.com only_moderator_can_post=no
```

```
python3 maillist_admin.py update all@ateinco.com moderated=no
```

quitar obligacion de moderar todas las listas

```
python3 maillist_admin.py update all@ateinco.com only_moderator_can_post=no
```

```
python3 maillist_admin.py update all@ateinco.com moderated=no
```