

Listmonk

Sistema de email marketing simple para un usuario

- [Instalar listmonk 3.0.0 en debian 12 con postgres 15](#)
- [Busqueda y filtrado de suscriptores en listmonk](#)

Instalar listmonk 3.0.0 en debian 12 con postgres 15

Ejecutamos el cliente de la base de datos postgres, como usuario postgres

```
sudo -u postgres psql
```

Creamos la base de datos el usuario y otorgamos permisos

```
create database listmonk;
create user listmonk with encrypted password 'yourpasswordhere';
grant all privileges on database listmonk to listmonk;
ALTER DATABASE listmonk OWNER TO listmonk;
\q
```

Luego descargamos la última versión de listmonk. Ver

```
wget
https://github.com/knadh/listmonk/releases/download/v3.0.0/listmonk_3.0.0_linux_amd64.tar.gz
tar -zxvf listmonk_3.0.0_linux_amd64.tar.gz
./listmonk --new-config
# Editamos con usuario/contraseña y datos usuario base de datos
nano config.toml
./listmonk --install
mv config.toml /etc/listmonk
mv /etc/listmonk /etc/listmonk.conf
mkdir /etc/listmonk
mv /etc/listmonk.conf /etc/listmonk/config.toml
mv listmonk /usr/bin/
```

Creamos el servicio listmonk editando el archivo correspondiente

```
# Creamos el servicio listmonk
nano /etc/systemd/system/listmonk.service
```

con los siguientes datos:

```

[Unit]
Description=listmonk mailing list and newsletter manager (%I)
ConditionPathExists=/etc/listmonk/config.toml
Wants=network.target
# The PostgreSQL database may not be on the same host but if it
# is listmonk should wait for it to start up.
#After=postgresql.service

[Service]
Type=simple
EnvironmentFile=-/etc/default/listmonk
EnvironmentFile=-/etc/default/listmonk-%i
ExecStartPre=/usr/bin/listmonk --config /etc/listmonk/config.toml --upgrade --yes
ExecStart=/usr/bin/listmonk --config /etc/listmonk/config.toml $SYSTEMD_LISTMONK_ARGS
Restart=on-failure

# Create dynamic users for listmonk service instances
# but create a state directory for uploads in /var/lib/private/%i.
DynamicUser=True
StateDirectory=listmonk-%i
Environment=HOME=%S/listmonk-%i
WorkingDirectory=%S/listmonk-%i

# Use systemd's ability to disable security-sensitive features
# that listmonk does not explicitly need.
# NoNewPrivileges should be enabled by DynamicUser=yes but systemd-analyze
# still recommended to explicitly enable it.
NoNewPrivileges=True

# listmonk doesn't need any capabilities as defined by the linux kernel
# see: https://man7.org/linux/man-pages/man7/capabilities.7.html
CapabilityBoundingSet=

# listmonk only executes native code with no need for any other ABIs.
SystemCallArchitectures=native
# Only enable a reasonable set of system calls.
# see: https://www.freedesktop.org/software/systemd/man/systemd.exec.html#SystemCallFilter
#SystemCallFilter=@system-service
#SystemCallFilter=~@privileged
# ProtectSystem=strict, which is implied by DynamicUser=True, already disabled write calls
# to the entire filesystem hierarchy, leaving only /dev/, /proc/, and /sys/ writable.
# listmonk doesn't need access to those so might as well disable them.

```

```
PrivateDevices=True
ProtectControlGroups=True
ProtectKernelTunables=True
# Make /home/, /root/, and /run/user/ inaccessible.
ProtectHome=True
# listmonk doesn't handle any specific device nodes.
DeviceAllow=False
# listmonk doesn't make use of linux namespaces.
RestrictNamespaces=True
# listmonk doesn't need realtime scheduling.
RestrictRealtime=True
# Make sure files created by listmonk are only readable by itself and
# others in the listmonk system group.
UMask=0027
# Disable memory mappings that are both writable and executable.
MemoryDenyWriteExecute=True
# listmonk doesn't make use of linux personality switching.
LockPersonality=True
# listmonk only needs to support the IPv4 and IPv6 address families.
RestrictAddressFamilies=AF_INET AF_INET6
# listmonk doesn't need to load any linux kernel modules.
ProtectKernelModules=True
# Create a sandboxed environment where the system users are mapped to a
# service-specific linux kernel namespace.
PrivateUsers=True

[Install]
WantedBy=multi-user.target
```

Luego habilitamos el servicio

```
systemctl daemon-reload
systemctl enable listmonk.service
systemctl start listmonk.service
systemctl status listmonk.service
```


Busqueda y filtrado de suscriptores en listmonk

Para buscar un suscriptor por cualquier parte de su nombre o correo, se utiliza el campo normal de búsqueda.. Para cualquier otra búsqueda, debe utilizarse el apartado "Avanzado".

Al hacer click en el ícono o la palabra "Avanzado", se abre un nuevo campo, donde podremos ingresar consultas, en un formato de lenguaje de base de datos (SQL).

La base de datos de Listmonk, posee campos fijos, y un campo de atributos, que puede contener cualquier información.

Los campos fijos de la base son los siguientes:

Campo	Descripción
<code>subscribers.uuid</code>	Una indentificación generada al azar del suscriptor
<code>subscribers.email</code>	El E-mail ID del suscriptor
<code>subscribers.name</code>	El nombre del suscriptor
<code>subscribers.status</code>	El estado del suscriptor: enabled, disabled o blocklisted (habilitado, deshabilitado, bloqueado)
<code>subscribers.attrs</code>	Una serie de atributos accesibles mediante el operador <code>-></code> y <code>->></code>
<code>subscribers.created_at</code>	Fecha y hora en que se agregó el suscriptor a la base
<code>subscribers.updated_at</code>	Fecha y hora en la que el suscriptor se modificó por última vez

Ejemplos de búsquedas avanzadas

Buscar un suscriptor por e-mail

```
-- Búsqueda exacta
subscribers.email = 'juanfernandez@empresa.com'

-- Busqueda parcial por terminación del email.
subscribers.email LIKE '%@empresa.com'
```

Buscar un suscriptor por nombre

```
-- Buscar todos los suscriptores cuyo nombre comienza con 'Juan'
subscribers.name LIKE 'Juan%'
```

Buscar los suscriptores bloqueados

```
-- Encontrar todos los suscriptores que han sido bloqueados.
subscribers.status = 'blocklisted'
```

Condiciones múltiples

```
-- Encontrar todos los 'Juan' que han sido bloqueados.
subscribers.email LIKE 'Juan%' AND subscribers.status = 'blocklisted'
```

Consultando por atributos

```
-- El operador ->> devuelve el valor del atributo como texto.
-- Encontrar todos los suscriptores cuyo pais es Argentina
subscribers.attrs->>'pais' = 'Argentina'
-- Encontrar todos los suscriptores cuyo pais es Chile
-- y son de educacion
subscribers.attrs->>'pais' = 'Chile' AND
    (subscribers.attrs->>'educacion') = '1'
-- Encontrar todos los suscriptores cuyo pais es Argentina
-- y NO son de educacion
subscribers.attrs->>'pais' = 'Argentina' AND
    (subscribers.attrs->>'educacion') != '1'
```

Una explicación de los filtros en inglés, puede encontrarse en <https://listmonk.app/docs/querying-and-segmentation/>