📁 LPIC-2 / 📨 Examen 211 - Servicios de Correo Electrónico - Ejercicios

*Nota: Estos ejercicios implican instalar software y modificar configuraciones que afectan al flujo de correo. Realízalos **SIEMPRE en una VM de prueba dedicada**. Asegúrate de que tu VM tiene acceso a internet para descargar actualizaciones. Necesitarás privilegios de superusuario (`sudo`).*

Ejercicio 11.3.1: Instalando Software de Filtrado (SpamAssassin y ClamAV)

- **Objetivo:** Instalar las herramientas comunes de filtrado de correo.
- **Requisitos:** Acceso a la línea de comandos. Privilegios de superusuario (`sudo`). Conexión a internet.
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Instala SpamAssassin:** `sudo apt update && sudo apt install spamassassin` (Debian/Ubuntu) o `sudo dnf install spamassassin` (Red Hat/CentOS/Fedora).
    3. **Instala ClamAV y el demonio:** `sudo apt install clamav clamav-daemon` (Debian/Ubuntu) o `sudo dnf install clamav clamav-daemon` (Red Hat/CentOS/Fedora).

Ejercicio 11.3.2: Gestión de Servicios de Filtrado

- **Objetivo:** Asegurarse de que los demonios de filtrado (`spamd`, `clamd`) están corriendo si se van a usar.
- **Requisitos:** Software de filtrado instalado. Privilegios de superusuario (`sudo`).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Habilita e inicia los servicios:**
        - SpamAssassin: `sudo systemctl enable spamd && sudo systemctl start spamd`.
        - ClamAV: `sudo systemctl enable clamav-daemon && sudo systemctl start clamav-daemon`.
    3. **Verifica el estado:** `systemctl status spamd.service clamav-daemon.service`. Deberían estar `active (running)`.

Ejercicio 11.3.3: Actualizando Reglas y Definiciones

- **Objetivo:** Asegurarse de que las herramientas de filtrado tienen las bases de datos más recientes.
- **Requisitos:** Software de filtrado instalado. Conexión a internet. Privilegios de superusuario (`sudo`).
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Actualiza las reglas de SpamAssassin:** Ejecuta `sudo sa-update`. Debería descargar y aplicar las actualizaciones si las hay.

3. **Actualiza las definiciones de virus de ClamAV:** Ejecuta `sudo freshclam`. Debería descargar y aplicar las actualizaciones. **Nota:** La primera ejecución puede tardar y descargar una base de datos grande.
4. **Verifica los logs de `freshclam`** (generalmente en `/var/log/clamav/freshclam.log` o journald) para confirmar que la actualización fue exitosa.

Ejercicio 11.3.4: Localizando Archivos de Configuración de Filtros

- **Objetivo:** Encontrar los directorios y archivos de configuración principales para SpamAssassin y ClamAV.
- **Requisitos:** Software de filtrado instalado. Acceso a la línea de comandos.
- **Desarrollo Paso a Paso:**
    1. Abre una terminal.
    2. **Explora el directorio de configuración de SpamAssassin:** Ejecuta `ls -l /etc/mail/spamassassin/`. Busca archivos como `local.cf`.
    3. **Explora el directorio de configuración de ClamAV:** Ejecuta `ls -l /etc/clamav/`. Busca archivos como `clamd.conf` y `freshclam.conf`.
    4. **Visualiza un archivo de configuración de ejemplo:** Ejecuta `sudo less /etc/mail/spamassassin/local.cf` o `sudo less /etc/clamav/clamd.conf`.

Ejercicio 11.3.5: (Conceptual) Integrando SpamAssassin con Postfix (content_filter)

- **Objetivo:** Entender cómo configurar Postfix para que envíe correos a SpamAssassin.
- *Requisitos:* Postfix y SpamAssassin instalados. Privilegios de superusuario (`sudo`). **VM de prueba.**
- **Desarrollo Paso a Paso (Conceptual):**
    1. Abre una terminal.
    2. **Edita el archivo de configuración principal de Postfix:** `sudo vi /etc/postfix/main.cf`.
    3. **Añade la directiva `content_filter` para apuntar a un servicio local (ej: `spamcheck`):**

        ```
        content_filter = spamcheck:127.0.0.1:10024
        ```

        - Esto le dice a Postfix que después de recibir un correo, lo envíe vía SMTP a la dirección `127.0.0.1` en el puerto `10024`. Necesitas configurar un servicio en ese puerto.
    4. **Edita el archivo `master.cf` de Postfix:** `sudo vi /etc/postfix/master.cf`.
    5. **Define el servicio `spamcheck` que recibirá el correo y lo pasará a `spamc`:**

        ```
        # Servicio para pasar correo a spamassassin
        spamcheck unix - n n - - smtp
        ```

```
    -o smtp_send_xforward_command=yes

127.0.0.1:10025 inet n n - - smtpd # Este es el servicio que recibe
DE spamassassin
    -o content_filter=
    -o
receive_override_options=no_header_body_checks,no_unknown_recipient_
checks,no_address_mappings
    -o smtpd_helo_restrictions=
    -o smtpd_client_restrictions=
    -o smtpd_sender_restrictions=
    -o smtpd_recipient_restrictions=permit_mynetworks,reject
    -o mynetworks=127.0.0.0/8,[::1]/128
    -o smtpd_data_restrictions=
    -o smtpd_end_of_data_restrictions=
    -o smtpd_restriction_classes=
    -o www_alias=
    -o smtpd_null_sender_reject=no
    -o smtpd_peername=unknown_hostname
    -o smtpd_helo_timeout=0
    -o smtpd_send_ehlo_command=yes
    -o smtp_bind_address=127.0.0.1
    -o smtp_bind_address6=::1
    -o smtp_address_preference=any
    -o disable_dns_lookups=yes

# Servicio real que llama a spamc - a menudo definido en un script
wrapper o con amavisd-new
# Este es un ejemplo simplificado, una integracion real es mas
compleja y usa scripts o amavisd-new
# 127.0.0.1:10024 inet n n - - pipe
#   flags=Rq user=filter argv=/usr/bin/spamc -f -e
/usr/sbin/sendmail -oi -f ${sender} ${recipient}
```

- **Nota:** La integración `content_filter` completa con SpamAssassin y ClamAV es compleja y a menudo se maneja mejor con un middleware como `amavisd-new`. El ejemplo anterior solo ilustra el concepto de `content_filter` y requiere que el servicio en 10024 exista y devuelva el correo filtrado al puerto 10025.

6. **Guarda y sal de ambos archivos (.cf).**
7. **Verifica la sintaxis de Postfix:** `sudo postfix check`.
8. **Recarga Postfix:** `sudo systemctl reload postfix`.

Ejercicio 11.3.6: (Conceptual) Integrando ClamAV con Postfix (Milter o Content Filter)

- **Objetivo:** Entender cómo configurar Postfix para que envíe correos a ClamAV.
- *Requisitos:* Postfix y ClamAV instalados. Privilegios de superusuario (`sudo`). **VM de prueba.**
- **Desarrollo Paso a Paso (Conceptual):**
  1. Abre una terminal.
  2. **Método Milter (usando `clamav-milter`):**
     - Instala el paquete `clamav-milter`.

- Edita el archivo de configuración de `clamav-milter` (a menudo en `/etc/clamav/clamav-milter.conf`) para apuntar al socket de `clamd` y configurar opciones.
- Edita `main.cf` de Postfix y añade la directiva `smtpd_milters`:

  ```
  smtpd_milters = unix:/var/spool/clamav/clamav-milter.sock # La ruta del socket puede variar
  ```

  (La ruta exacta del socket se configura en `clamav-milter.conf` y en la unidad de systemd de `clamav-milter`).
- Habilita e inicia el servicio `clamav-milter.service`.
- Recarga Postfix.

3. **Método Content Filter (usando un wrapper, ej: con `amavisd-new`):**
   - Instala el paquete `amavisd-new` (que a menudo depende de SpamAssassin y ClamAV o los recomienda).
   - Configura `amavisd-new` (su archivo de configuración principal es grande, a menudo `/etc/amavis/conf.d/15-content_filter_mode`). Amavisd-new se configura para llamar a `spamc` y `clamscan` (o interactuar con los demonios).
   - Configura Postfix para que el `content_filter` apunte al servicio de `amavisd-new` (Amavisd-new se configura para escuchar en un puerto, ej: 10024, y enviar el correo procesado de vuelta a Postfix en otro puerto, ej: 10025). Modifica `main.cf` y `master.cf` para integrar con los puertos que Amavisd-new escucha.
   - Habilita e inicia el servicio `amavisd-new.service`.
   - Recarga Postfix.

Ejercicio 11.3.7: (Conceptual) Configurando Restricciones Básicas de Postfix

- **Objetivo:** Entender cómo usar las reglas de Postfix para filtrar correo sin herramientas externas.
- *Requisitos:* Postfix instalado. Privilegios de superusuario (`sudo`). **VM de prueba.**
- **Desarrollo Paso a Paso (Conceptual):**
  1. Abre una terminal.
  2. **Edita `main.cf`:** `sudo vi /etc/postfix/main.cf`.
  3. **Añade o modifica las directivas de restricciones (Ejemplos):**
     - Rechazar correo de un dominio específico:

       ```
       smtpd_sender_restrictions =
          permit_mynetworks,
          reject_sender <lista_dominios_bloqueados>
       ```

       (Donde `<lista_dominios_bloqueados>` es un archivo con una lista de dominios, ej: `hash:/etc/postfix/sender_access`).

- Rechazar correo a destinatarios desconocidos (para dominios listados en `mydestination`):

```
smtpd_recipient_restrictions =
   permit_mynetworks,
   reject_unauth_destination, # Rechazar si no es para
misdominios o no soy relay
   reject_unknown_recipient_domain, # Rechazar si el dominio
del destinatario no tiene MX
   reject_non_canonical_sender # Opcional, requiere
reescritura de direcciones
   # Opciones para control de destinatarios:
   # reject_unverified_recipient # Verificar si el
destinatario existe (requiere mas config)
   # check_recipient_access hash:/etc/postfix/recipient_access
# Bloquear/permitir destinatarios
```

- Rechazar clientes de ciertas listas negras de DNS (RBLs):

```
smtpd_client_restrictions =
   permit_mynetworks,
   reject_rbl_client zen.spamhaus.org # Ejemplo popular de RBL
```

4. **Guarda y sal.**
5. **Si creaste archivos de lookup como `sender_access` o `recipient_access`, debes convertirlos a base de datos:** `sudo postmap /etc/postfix/sender_access` (crea `sender_access.db`).
6. **Verifica la sintaxis:** `sudo postfix check`.
7. **Recarga Postfix:** `sudo systemctl reload postfix`.