Lo que tienes que hacer para crear esa política de asunto que solo afecte a un dominio (safecomercial.com), cómo representarla en el formato que la API de Zoho espera y qué endpoints debes usar. También indico opciones para que se envíe una copia a soporte@sfisas.com cuando alguien intente enviar un correo que no cumpla la regla.

Resumen rápido:

1. Consigue acceso API (OAuth) con un admin.
2. Crea una *policy* (política) y una *email restriction* (restricción de asunto).
3. Asocia la restricción a la policy.
4. Aplica la policy a los usuarios del dominio safecomercial.com (no hay un endpoint público que “aplique directamente a un dominio”, así que se hace seleccionando los usuarios del dominio).
5. Prueba y ajusta el comportamiento (rechazo + mensaje; ver opciones para copia a soporte).  
   (Las URLs y pasos abajo con ejemplos curl y JSON). [Zoho+4Zoho+4Zoho+4](https://www.zoho.com/mail/help/api/organization-api.html?utm_source=chatgpt.com)

**Requisitos previos (antes de tocar la API)**

1. Cuenta de administrador en Zoho Mail para la organización.
2. Registrar una aplicación OAuth en Zoho (obtener client\_id y client\_secret) y generar un **refresh token / access token** con scopes administrativos (al menos scopes relacionados con ZohoMail.organization.policy y los scopes para leer cuentas/usuarios). (La guía de inicio del API describe cómo obtener tokens). [Zoho](https://www.zoho.com/mail/help/api/getting-started-with-api.html?utm_source=chatgpt.com)
3. Tener a mano el **ZOID** de la organización (lo trae la API de Organization). [Zoho](https://www.zoho.com/mail/help/api/organization-api.html?utm_source=chatgpt.com)

**Endpoints clave (resumen)**

* Obtener detalles de la organización (obtener zoid):  
  GET https://mail.zoho.com/api/organization/ (header Authorization:Zoho-oauthtoken {access\_token}). [Zoho](https://www.zoho.com/mail/help/api/organization-api.html?utm_source=chatgpt.com)
* Crear una *policy* (política):  
  POST https://mail.zoho.com/api/organization/{zoid}/policy (body JSON con policyName). [Zoho](https://www.zoho.com/mail/help/api/?utm_source=chatgpt.com)
* Crear una *email restriction* (restricción de correo):  
  POST https://mail.zoho.com/api/organization/{zoid}/policy/mailRestriction (la documentación muestra cómo listar/crear restricciones). [Zoho](https://www.zoho.com/mail/help/api/get-email-restrictions.html?utm_source=chatgpt.com)
* Asignar una restricción a una *policy*:  
  PUT https://mail.zoho.com/api/organization/{zoid}/policy/{policyId} (usar el modo adecuado para “assign / change mail restriction”). [Zoho](https://www.zoho.com/mail/help/api/assign-email-restriction-to-policy.html?utm_source=chatgpt.com)
* Obtener todos los usuarios (para filtrar por dominio):  
  GET https://mail.zoho.com/api/organization/{zoid}/accounts → filtrar por email que termina en @safecomercial.com. [Zoho](https://www.zoho.com/mail/help/api/users-api.html?utm_source=chatgpt.com)
* Aplicar la policy a usuarios (una vez tengas la lista de zuid):  
  PUT https://mail.zoho.com/api/organization/{zoid}/policy/{policyId} con mode=applyPolicyToUsers y userList = array de zuid de esos usuarios. [Zoho](https://www.zoho.com/mail/help/api/apply-policy-to-users-or-groups.html?utm_source=chatgpt.com)

Nota: la consola de administración explica que cuando una política de asunto bloquea un correo, **el servidor saliente lo rechaza y hace bounce** al remitente (puedes añadir mensaje de rechazo). Para forzar copia o comportamiento adicional hay que usar las acciones soportadas o una solución alternativa si la API no expone exactamente lo que quieres. [Zoho](https://www.zoho.com/mail/help/adminconsole/email-policy.html)

**Paso a paso con ejemplos (lo más práctico)**

**0) Generar tokens (OAuth)**

* Si ya tienes un refresh\_token, obtén access\_token así:

curl -X POST "https://accounts.zoho.com/oauth/v2/token" \

-d "client\_id=TU\_CLIENT\_ID" \

-d "client\_secret=TU\_CLIENT\_SECRET" \

-d "grant\_type=refresh\_token" \

-d "refresh\_token=TU\_REFRESH\_TOKEN"

* Usa el access\_token recibido en los headers:  
  Authorization: Zoho-oauthtoken {access\_token}. (Si no tienes refresh token, usa el flujo de autorización o el "self-client" en Zoho para generar uno). [Zoho](https://www.zoho.com/mail/help/api/getting-started-with-api.html?utm_source=chatgpt.com)

**1) Obtener el ZOID de la organización**

curl -H "Authorization:Zoho-oauthtoken {access\_token}" \

"https://mail.zoho.com/api/organization/"

Guarda el zoid que te devuelva.

(Referencia API Organization). [Zoho](https://www.zoho.com/mail/help/api/organization-api.html?utm_source=chatgpt.com)

**2) Crear la policy (política donde se aplicarán las restricciones)**

curl -X POST "https://mail.zoho.com/api/organization/{zoid}/policy" \

-H "Authorization:Zoho-oauthtoken {access\_token}" \

-H "Content-Type: application/json" \

-d '{"policyName":"Validación de Códigos de Proyecto y Documento (safecomercial.com)"}'

Guarda el policyId de la respuesta. [Zoho](https://www.zoho.com/mail/help/api/?utm_source=chatgpt.com)

**3) Crear la email restriction (restricción de asunto)**

La documentación pública muestra el endpoint POST /api/organization/{zoid}/policy/mailRestriction para crear restricciones. En algunos casos la documentación solo muestra crear la restricción por restrictionName y luego asociarla a la policy; la consola web permite definir con detalle *dominios / sujetos / acciones*. Por eso hay dos caminos (te explico ambos):

**Opción A — intentar crear la restricción con el JSON completo vía API (si tu cuenta y la API la aceptan):**

Usa un payload JSON con la estructura lógica (AND/OR) que ya escribiste — esto fuerza que se cumplan **ambos** tipos (obraX **y** codY) **en el mismo asunto**. Aquí tienes el ejemplo (ajusté name → restrictionName porque la API suele usar ese campo; si falla intenta name):

{

"restrictionName": "Validación de Códigos de Proyecto y Documento (safecomercial.com)",

"scope": "outgoing",

"conditions": {

"logical\_operator": "AND",

"sub\_conditions": [

{

"logical\_operator": "AND",

"expressions": [

{

"field": "From",

"operator": "ends with",

"value": "safecomercial.com"

}

]

},

{

"logical\_operator": "AND",

"sub\_conditions": [

{

"logical\_operator": "OR",

"expressions": [

{"field":"Subject","operator":"contains","value":"obra1"},

{"field":"Subject","operator":"contains","value":"obra2"},

{"field":"Subject","operator":"contains","value":"obra3"},

{"field":"Subject","operator":"contains","value":"obra4"}

]

},

{

"logical\_operator": "OR",

"expressions": [

{"field":"Subject","operator":"contains","value":"cod1"},

{"field":"Subject","operator":"contains","value":"cod2"},

{"field":"Subject","operator":"contains","value":"cod3"},

{"field":"Subject","operator":"contains","value":"cod4"}

]

}

]

}

]

},

"actions": [

{

"action": "reject\_mail",

"message": "Atención, mensaje no enviado porque UD olvidó colocar los códigos en el asunto. Corrija y vuelva a enviar."

},

{

"action": "cc\_to\_mail",

"email\_address": "soporte@sfisas.com"

}

]

}

Ejemplo curl (intenta primero con esto; si la API acepta atributos de condiciones, funcionará):

curl -X POST "https://mail.zoho.com/api/organization/{zoid}/policy/mailRestriction" \

-H "Authorization:Zoho-oauthtoken {access\_token}" \

-H "Content-Type: application/json" \

-d @payload.json

**Opción B — si la API devuelve error porque sólo acepta restrictionName al crear:**

1. Crea la restricción con sólo restrictionName vía API.
2. Entra al Admin Console (Mail Settings → Email Restrictions) y edita la restricción para pegar las condiciones y acciones (desde la UI es 100% seguro). La documentación del Admin Console muestra cómo construir las reglas y confirma que si un correo no cumple la condición, es rechazado y se le hace bounce al remitente (con mensaje). [Zoho+1](https://www.zoho.com/mail/help/api/get-email-restrictions.html?utm_source=chatgpt.com)

En muchos despliegues la creación completa de condiciones complejas se hace desde la consola web; la API permite creación/consulta/asignación; prueba la Opción A primero y si recibes 400 o un error, usa la B (creación mínima por API + configuración por UI).

**4) Asociar la restricción a la *policy***

Una vez que tengas el restrictionID (por ejemplo desde GET /api/organization/{zoid}/policy/mailRestriction), asigna esa restricción a la policy:

curl -X PUT "https://mail.zoho.com/api/organization/{zoid}/policy/{policyId}" \

-H "Authorization:Zoho-oauthtoken {access\_token}" \

-H "Content-Type: application/json" \

-d '{

"mode":"changeMailRestrictionID",

"orgMailRestrictionID":[ "1003505000077...." ]

}'

(El mode exacto para assign puede variar; la documentación de “Assign Email Restriction to Policy” describe el endpoint y el uso. Si tu llamada requiere una clave diferente, la respuesta te lo indicará). [Zoho](https://www.zoho.com/mail/help/api/assign-email-restriction-to-policy.html?utm_source=chatgpt.com)

**5) Aplicar la policy solo al dominio safecomercial.com**

Zoho no documenta un endpoint “aplicar política a dominio directamente”; la forma recomendada es:

1. Obtener todos los usuarios / cuentas de la organización:  
   GET https://mail.zoho.com/api/organization/{zoid}/accounts → del resultado filtras por email que termine en @safecomercial.com. [Zoho](https://www.zoho.com/mail/help/api/users-api.html?utm_source=chatgpt.com)
2. Con la lista de zuid (identificadores de usuario), haces:

curl -X PUT "https://mail.zoho.com/api/organization/{zoid}/policy/{policyId}" \

-H "Authorization:Zoho-oauthtoken {access\_token}" \

-H "Content-Type: application/json" \

-d '{

"mode":"applyPolicyToUsers",

"userList": [ 123456789012345, 123456789012346, ... ]

}'

Así la policy se aplica a todos los usuarios del dominio. [Zoho](https://www.zoho.com/mail/help/api/apply-policy-to-users-or-groups.html?utm_source=chatgpt.com)

**6) Comprobaciones / pruebas**

* Envía desde un usuario @safecomercial.com un correo cuyo asunto **sí** contiene obra1 y cod1 → debe enviarse.
* Envía un correo **sin** alguno de los dos tokens → servidor saliente debería **rechazar** y generar bounce con el mensaje que definiste en reject\_mail. (La consola de administración confirma este comportamiento de rechazo/bounce). [Zoho](https://www.zoho.com/mail/help/adminconsole/email-policy.html)

**¿Y la copia a soporte@sfisas.com si alguien incumple?**

* En tu JSON pusiste cc\_to\_mail — **puede que funcione** si la implementación de la API de Zoho en tu tenant acepta esa acción en el actions del restriction. Prueba la Opción A (enviar el JSON completo).
* Si la API no acepta cc\_to\_mail dentro de la restricción, alternativas prácticas:
  1. **Mensaje personalizado de rechazo**: Ya puedes configurar reject\_mail con un texto claro (el remitente ve por qué falló). Esto es seguro y documentado. [Zoho](https://www.zoho.com/mail/help/adminconsole/email-policy.html)
  2. **Forward/Journal a soporte**: crear una *Forward Restriction* o usar una regla de auditoría/journaling para copiar todos los correos salientes (o los rechazados, si ese caso está disponible) a una cuenta interna. Nota: *forward restriction* suele aplicarse a TODO el tráfico saliente (no sólo a violaciones). Si quieres **solo copias de incidencias**, puede que necesites un proceso externo (webhook/journal) que detecte los bounces y los re-envíe a soporte@sfisas.com.
  3. **Proceso externo**: usar un mailbox de administración que reciba los mensajes bounce y reenviarlos (o registrar el incidente).
* Conclusión: primero prueba si cc\_to\_mail es aceptado (enviando el payload). Si no, lo más simple es usar reject\_mail + mensaje y, si hace falta copia, montar un proceso de auditoría/forwarding o un pequeño script que detecte bounces y notifique a soporte.

**Ejemplo práctico completo (resumen de comandos)**

1. Obtener access\_token (mediante refresh\_token).
2. Obtener zoid.
3. Crear policy (guardar policyId).
4. Intentar crear mailRestriction con el JSON completo (guardar restrictionID); si da error crea con restrictionName y edítalo en la Consola.
5. Asignar restriction a policy (PUT).
6. Obtener usuarios, filtrar por dominio y aplicar policy a esos zuid (PUT con mode=applyPolicyToUsers).
7. Probar envío. (Si necesitas help para cualquiera de los curl con placeholders, te los doy listos para copiar.)

(Ya te mostré los curl y el JSON arriba — úsalos con tus valores.)

**Advertencias y consejos finales**

* **Plan Email Lite y acceso API**: en general la API de organización/policy se controla a nivel admin; si tienes una suscripción “Mail Lite” deberías poder usar las APIs de admin si eres administrador, pero algunos features avanzados (journaling, archiving, etc.) pueden necesitar planes superiores. Si un endpoint te devuelve 403 o 401 comprueba scopes y que tu cuenta/plan tenga permisos. (Si quieres, reviso tu caso concreto si me pegas el error). [Zoho](https://www.zoho.com/mail/help/api/getting-started-with-api.html?utm_source=chatgpt.com)
* Si la API no acepta las condiciones completas, la vía segura es la **consola Admin → Email Restrictions** para dejar la regla exactamente como la necesitas, y usar la API sólo para asignar/automatizar la aplicación a usuarios. [Zoho](https://www.zoho.com/mail/help/adminconsole/email-policy.html)

Si quieres, hago **una de estas tres cosas ahora mismo** (elige una y lo hago en esta respuesta):

1. Te devuelvo el JSON final listo para POST (con restrictionName o con name, ambos), y los curl completos con placeholders rellenados.
2. Te genero un pequeño script (Python) que: toma listas obras = [...] y cods = [...] y crea el JSON compacto (con OR dentro de AND) listo para enviar por API.
3. Te doy el flujo exacto para obtener refresh\_token desde la consola de Zoho (paso a paso) si nunca lo hiciste.