

Guía de Implementación: Sección de Notas Técnicas

Sistema de Blog para Procedimientos Operativos en Astro + Starlight

Índice

1. [Visión General del Sistema](#)
 2. [Arquitectura de la Solución](#)
 3. [Implementación Paso a Paso](#)
 4. [Estructura de Archivos](#)
 5. [Configuración Avanzada](#)
 6. [Casos de Uso y Ejemplos](#)
-

Visión General del Sistema

Contexto Técnico

Tu sitio actualmente utiliza:

- **Astro 5.17.1** con el framework **Starlight 0.37.5**
- **Content Collections** con el nuevo sistema de `(loaders)` de Astro 5
- Soporte multiidioma (español/inglés) mediante `(locales)`
- Estructura organizada en subdirectorios: `(linux-ops)`, `(az-cloud-ops)`, `(projects)`

Objetivo de la Implementación

Crear una **sección de Notas Técnicas** (Tech Notes) que funcione como un blog de procedimientos operativos, manteniendo la filosofía de SOPs atómicos y reutilizables, con las siguientes características:

- Colección separada para facilitar la gestión
- Soporte completo para i18n (español/inglés)
- Metadata enriquecida (fechas, tags, autor, última actualización)
- Página de índice cronológico automático
- Feed RSS integrado
- Compatibilidad total con la estructura existente

Arquitectura de la Solución

Decisión de Diseño: ¿Plugin o Colección Nativa?

Opción Recomendada: Colección Nativa Personalizada

Razones Técnicas:

1. **Control Total del Schema:** Puedes definir metadatos específicos para SOPs (autor, sistema operativo, versión, criticidad, etc.)
2. **Sin Dependencias Externas:** No dependes de actualizaciones de plugins de terceros
3. **Integración Natural:** Se mantiene la coherencia con tu actual `content.config.ts`
4. **Rendimiento:** No añades peso adicional al bundle
5. **Flexibilidad:** Puedes crear layouts y componentes personalizados

Por qué NO usar `@astrojs/starlight-blog`:

- Está diseñado para blogs de contenido, no para documentación técnica operativa
 - Añade opiniones sobre estructura que pueden entrar en conflicto con tus necesidades
 - Requiere adaptaciones adicionales para multiidioma en contexto de SOPs
-

Implementación Paso a Paso

PASO 1: Actualizar `src/content.config.ts`

Reemplaza tu archivo actual con esta versión mejorada:

```
typescript
```

```

// src/content.config.ts

import { defineCollection, z } from 'astro:content';
import { docsSchema } from '@astrojs/starlight/schema';
import { glob } from 'astro/loaders';

// =====
// 1. COLECCIÓN: docs (Documentación Principal)
// =====

const docs = defineCollection({
  loader: glob({ pattern: '**/*.{md,mdx}', base: 'src/content/docs' }),
  schema: docsSchema({
    extend: z.object({
      source: z.string().optional(),
      domain: z.enum([
        'compute', 'storage', 'networking', 'security',
        'virtualization', 'containers', 'automation',
        'monitoring', 'cloud'
      ]).optional(),
      type: z.enum([
        'lab', 'theory', 'cheatsheet', 'scenario',
        'configuration', 'project'
      ]).default('theory'),
    }),
  }),
});

// =====
// 2. COLECCIÓN: tech-notes (Notas Técnicas / Blog)
// =====

const techNotes = defineCollection({
  loader: glob({ pattern: '**/*.{md,mdx}', base: 'src/content/tech-notes' }),
  schema: z.object({
    // Metadatos básicos del artículo
    title: z.string(),
    description: z.string(),
    date: z.coerce.date(), // Fecha de publicación (formato: YYYY-MM-DD)
    updated: z.coerce.date().optional(), // Última actualización

    // Clasificación y contexto
    tags: z.array(z.string()).default([]), // Ejemplo: ['bash', 'networking', 'troubleshooting']
    category: z.enum([
      'sop', // Standard Operating Procedure
      'troubleshooting', // Resolución de incidentes
    ])
  })
});

```

```
'howto',      // Guía de procedimiento
'quickref',    // Referencia rápida
'case-study', // Caso de estudio
'automation'  // Scripts y automatización
]).default('sop'),

// Metadatos técnicos
systems: z.array(z.enum([
  'linux', 'windows', 'azure', 'aws', 'gcp',
  'kubernetes', 'docker', 'ansible', 'terraform'
])).optional(), // Sistemas operativos o plataformas relevantes

difficulty: z.enum(['beginner', 'intermediate', 'advanced']).default('intermediate'),

// Información del autor y estado
author: z.string().default('Daniel Zamo'),
draft: z.boolean().default(false), // Para borradores no publicados

// SEO y búsqueda
keywords: z.array(z.string()).optional(),

// Idioma (importante para i18n)
lang: z.enum(['es', 'en']).default('es'),
),

});

=====

// 3. COLECCIÓN: questions (Quizzes)
=====

const questions = defineCollection({
  loader: glob({ pattern: '**/*.json,yaml', base: 'src/content/questions' }),
  schema: z.object({
    exam: z.string(),
    topic: z.string(),
    difficulty: z.enum(['easy', 'medium', 'hard']),
    question: z.object({
      scenario: z.string().optional(),
      prompt: z.string(),
    }),
    options: z.array(z.object({ id: z.string(), text: z.string() })),
    correctAnswerId: z.string(),
    explanation: z.object({
      summary: z.string(),
      breakdown: z.array(z.object({ optionId: z.string(), reasoning: z.string() })),
    })
  })
});
```

```
    }),

    });

});

// =====
// EXPORTACIÓN DE COLECCIONES
// =====

export const collections = {
  docs,
  'tech-notes': techNotes,
  questions
};
```

¿Qué hemos añadido?

- Nueva colección `tech-notes` con schema específico para procedimientos operativos
 - Metadatos enriquecidos: fecha, tags, categoría, sistemas, dificultad
 - Campo `lang` para gestión explícita de i18n
 - Campo `draft` para trabajar en borradores sin publicar
-

PASO 2: Crear Estructura de Directorios

Ejecuta estos comandos desde la raíz de tu proyecto:

```
bash

# Crear directorio principal para tech-notes
mkdir -p src/content/tech-notes/{es,en}

# Crear subdirectorios por categoría (opcional pero recomendado)
mkdir -p src/content/tech-notes/es/{sop,troubleshooting,howto}
mkdir -p src/content/tech-notes/en/{sop,troubleshooting,howto}

# Crear directorio para la página de índice del blog
mkdir -p src/pages/tech-notes
```

Estructura resultante:

```
src/content/
├── docs/      # Documentación principal (ya existe)
│   ├── en/
│   └── es/
├── tech-notes/ # NUEVA: Notas técnicas
│   ├── en/
│   │   ├── sop/
│   │   ├── troubleshooting/
│   │   └── howto/
│   └── es/
│       ├── sop/
│       ├── troubleshooting/
│       └── howto/
└── questions/  # Quizzes (ya existe)
```

PASO 3: Migrar las Notas Existentes

Tienes 3 notas en `[src/content/docs/{en,es}/notes/]`. Vamos a migrarlas a la nueva colección:

```
bash
```

```

# Mover notas en inglés
mv src/content/docs/en/notes/wsl-export-guide.mdx \
src/content/tech-notes/en/howto/

mv src/content/docs/en/notes/wsl-migration-backup-strategy.mdx \
src/content/tech-notes/en/sop/

mv src/content/docs/en/notes/wsl-restoration-guide.mdx \
src/content/tech-notes/en/howto/

# Mover notas en español
mv src/content/docs/es/notes/wsl-export-guide.mdx \
src/content/tech-notes/es/howto/

mv src/content/docs/es/notes/wsl-migration-backup-strategy.mdx \
src/content/tech-notes/es/sop/

mv src/content/docs/es/notes/wsl-restoration-guide.mdx \
src/content/tech-notes/es/howto/

# Eliminar carpetas vacías
rmdir src/content/docs/en/notes
rmdir src/content/docs/es/notes

```

Actualizar Frontmatter de las Notas

Abre cada archivo (.mdx) migrado y actualiza el frontmatter. Ejemplo para `wsl-export-guide.mdx`:

```

yaml

---
title: "WSL Export Guide - Complete Backup Procedure"
description: "Step-by-step procedure to export and backup WSL distributions using tar archives"
date: 2026-01-15
updated: 2026-01-31
tags: ['wsl', 'windows', 'backup', 'linux']
category: 'howto'
systems: ['windows', 'linux']
difficulty: 'beginner'
author: 'Daniel Zamo'
lang: 'en'
keywords: ['wsl', 'export', 'backup', 'windows subsystem for linux']

---
```

PASO 4: Crear Página de Índice del Blog

Crea el archivo `(src/pages/tech-notes/index.astro)`:

```
astro
```

```
--  
// src/pages/tech-notes/index.astro  
import { getCollection } from 'astro:content';  
import StarlightPage from '@astrojs/starlight/components/StarlightPage.astro';  
  
// Obtener todas las notas técnicas  
const allNotes = await getCollection('tech-notes', ({ data }) => {  
    // Filtrar borradores en producción  
    return import.meta.env.PROD ? !data.draft : true;  
});  
  
// Agrupar por idioma  
const notesByLang = {  
    es: allNotes.filter(note => note.data.lang === 'es'),  
    en: allNotes.filter(note => note.data.lang === 'en')  
};  
  
// Ordenar por fecha (más reciente primero)  
notesByLang.es.sort((a, b) => b.data.date.valueOf() - a.data.date.valueOf());  
notesByLang.en.sort((a, b) => b.data.date.valueOf() - a.data.date.valueOf());  
  
// Detectar idioma actual (simplificado)  
const currentLang = Astro.url.pathname.includes('/en/') ? 'en' : 'es';  
const notes = notesByLang[currentLang];  
  
// Traducciones  
const translations = {  
    es: {  
        pageTitle: 'Notas Técnicas',  
        pageDescription: 'Procedimientos operativos, SOPs y documentación de incidentes en SysAdmin & DevOps',  
        readMore: 'Leer más',  
        publishedOn: 'Publicado el',  
        updatedOn: 'Actualizado el',  
        tags: 'Etiquetas',  
        noNotes: 'No hay notas publicadas aún.'  
    },  
    en: {  
        pageTitle: 'Technical Notes',  
        pageDescription: 'Operational procedures, SOPs and incident documentation in SysAdmin & DevOps',  
        readMore: 'Read more',  
        publishedOn: 'Published on',  
        updatedOn: 'Updated on',  
        tags: 'Tags',  
    }  
};
```

```
    noNotes: 'No published notes yet.'
  }
};

const t = translations[currentLang];
---

<StarlightPage
  frontmatter={{
    title: t.pageTitle,
    description: t.pageDescription,
  }}
>
<div class="tech-notes-index">
  {notes.length === 0 ? (
    <p>{t.noNotes}</p>
  ) : (
    <div class="notes-grid">
      {notes.map((note) => {
        const formattedDate = note.data.date.toLocaleDateString(currentLang, {
          year: 'numeric',
          month: 'long',
          day: 'numeric'
        });
        const updatedDate = note.data.updated?.toLocaleDateString(currentLang, {
          year: 'numeric',
          month: 'long',
          day: 'numeric'
        });
        return (
          <article class="note-card">
            <div class="note-header">
              <h2>
                <a href={`/ ${currentLang}/tech-notes/${note.id.replace(new RegExp(`^(es|en)/`), "")}`}>
                  {note.data.title}
                </a>
              </h2>
            <div class="note-meta">
              <time datetime={note.data.date.toISOString()}>
                {t.publishedOn} {formattedDate}
              </time>
              {note.data.updated && (

```

```

<time datetime={note.data.updated.toISOString()} class="updated">
  {t.updatedOn} {updatedDate}
</time>
)}
</div>
</div>

<p class="note-description">{note.data.description}</p>

{note.data.tags && note.data.tags.length > 0 && (
<div class="note-tags">
  <span class="tags-label">{t.tags}</span>
  {note.data.tags.map(tag => (
    <span class="tag">{tag}</span>
  )));
</div>
)}

<div class="note-footer">
  <span class="category-badge">{note.data.category}</span>
  <span class="difficulty-badge">{note.data.difficulty}</span>
</div>
</article>
);

})}
</div>
)
</div>
</StarlightPage>

<style>
.tech-notes-index {
  max-width: 1200px;
  margin: 0 auto;
  padding: 2rem 1rem;
}

.notes-grid {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(350px, 1fr));
  gap: 2rem;
  margin-top: 2rem;
}

```

```
.note-card {  
background: var(--sl-color-gray-6);  
border: 1px solid var(--sl-color-gray-5);  
border-radius: 8px;  
padding: 1.5rem;  
transition: transform 0.2s, box-shadow 0.2s;  
}  
  
}
```

```
.note-card:hover {  
transform: translateY(-4px);  
box-shadow: 0 8px 16px rgba(0, 0, 0, 0.3);  
}  
  
}
```

```
.note-header h2 {  
margin: 0 0 0.5rem 0;  
font-size: 1.5rem;  
}  
  
}
```

```
.note-header h2 a {  
color: var(--sl-color-white);  
text-decoration: none;  
}  
  
}
```

```
.note-header h2 a:hover {  
color: var(--sl-color-accent);  
}  
  
}
```

```
.note-meta {  
display: flex;  
flex-direction: column;  
gap: 0.25rem;  
font-size: 0.875rem;  
color: var(--sl-color-gray-3);  
margin-bottom: 1rem;  
}  
  
}
```

```
.note-meta .updated {  
font-style: italic;  
}  
  
}
```

```
.note-description {  
color: var(--sl-color-gray-2);  
line-height: 1.6;  
margin-bottom: 1rem;
```

```
}
```

```
.note-tags {  
  display: flex;  
  flex-wrap: wrap;  
  gap: 0.5rem;  
  margin-bottom: 1rem;  
  font-size: 0.875rem;  
}
```

```
.tags-label {  
  color: var(--sl-color-gray-3);  
  font-weight: bold;  
}
```

```
.tag {  
  background: var(--sl-color-gray-5);  
  color: var(--sl-color-accent);  
  padding: 0.25rem 0.75rem;  
  border-radius: 4px;  
  font-weight: 500;  
}
```

```
.note-footer {  
  display: flex;  
  gap: 0.75rem;  
  margin-top: 1rem;  
  padding-top: 1rem;  
  border-top: 1px solid var(--sl-color-gray-5);  
}
```

```
.category-badge,  
.difficulty-badge {  
  display: inline-block;  
  padding: 0.25rem 0.75rem;  
  border-radius: 4px;  
  font-size: 0.75rem;  
  font-weight: bold;  
  text-transform: uppercase;  
}
```

```
.category-badge {  
  background: var(--sl-color-accent);  
  color: var(--sl-color-black);
```

```
}

.difficulty-badge {
  background: var(--sl-color-gray-5);
  color: var(--sl-color-white);
}

@media (max-width: 768px) {
  .notes-grid {
    grid-template-columns: 1fr;
  }
}

</style>
```

PASO 5: Actualizar `astro.config.mjs`

Modifica el archivo `astro.config.mjs` para añadir un enlace al índice de notas técnicas:

```
javascript
```

```
// astro.config.mjs
import { defineConfig } from 'astro/config';
import starlight from '@astrojs/starlight';
import mermaid from 'astro-mermaid';

export default defineConfig({
  site: 'https://daniel-zamo.github.io',
  redirects: {
    '/': '/es/',
  },
  integrations: [
    mermaid(),
    starlight({
      title: 'DZ | SysAdmin & DevOps',
      favicon: '/favicon.png',
      social: [
        { icon: 'github', label: 'GitHub', href: 'https://github.com/daniel-zamo/daniel-zamo.github.io' },
        { icon: 'linkedin', label: 'Linkedin', href: 'https://www.linkedin.com/in/danielzamo/' }
      ],
      customCss: ['./src/styles/custom.css'],
      defaultLocale: 'es',
      locales: {
        en: { label: 'English', lang: 'en' },
        es: { label: 'Español', lang: 'es' }
      },
      head: [
        {
          tag: 'link',
          attrs: {
            rel: 'alternate',
            type: 'application/rss+xml',
            title: 'DZ | Tech Notes RSS Feed',
            href: '/rss.xml',
          },
        },
        {
          tag: 'script',
          content: `

```

```
const storedTheme = localStorage.getItem('starlight-theme');
if (!storedTheme) {
    document.documentElement.setAttribute('data-theme', 'dark');
    localStorage.setItem('starlight-theme', 'dark');
}
`,
},
],
sidebar: [
    // ⭐ NUEVO: Enlace a índice de Tech Notes
    {
        label: 'Tech Notes',
        translations: { es: 'Notas Técnicas' },
        link: 'tech-notes/',
        badge: { text: 'Blog', variant: 'note' }
    },
    // Resto de secciones existentes...
    {
        label: 'Azure Cloud Ops.',
        translations: { es: 'Operaciones Cloud Azure' },
        items: [
            {
                label: 'Introduction',
                link: 'az-cloud-ops/',
                translations: { es: 'Introducción' }
            },
            {
                label: 'Compute',
                translations: { es: 'Cómputo' },
                autogenerate: { directory: 'az-cloud-ops/compute' },
                collapsed: true
            },
            {
                label: 'Networking',
                translations: { es: 'Redes' },
                autogenerate: { directory: 'az-cloud-ops/networking' },
                collapsed: true
            },
        ],
    },
],
```

```
label: 'Projects',
translations: { es: 'Proyectos' },
items: [
  { label: 'Introduction', link: 'projects/', translations: { es: 'Introducción' } },
  { label: 'Automation', translations: { es: 'Automatización' }, autogenerate: { directory: 'projects/automation' } },
  { label: 'Middleware', translations: { es: 'Middleware' }, autogenerate: { directory: 'projects/middleware' } },
],
},
],
}),
],
});
```

Cambios clave:

- Eliminada la sección `(autogenerate: { directory: 'notes' })` (ya no existe)
 - Añadido enlace directo a `(/tech-notes/)` con badge visual
 - Mantiene toda la configuración i18n existente
-

PASO 6: Actualizar Feed RSS

Modifica `(src/pages/rss.xml.js)` para usar la nueva colección:

```
javascript
```

```
// src/pages/rss.xml.js
import rss from '@astrojs/rss';
import { getCollection } from 'astro:content';

export async function GET(context) {
    // Obtener todas las notas técnicas (sin borradores en producción)
    const techNotes = await getCollection('tech-notes', ({ data }) => {
        return import.meta.env.PROD ? !data.draft : true;
    });

    // Ordenar por fecha descendente
    const sortedNotes = techNotes.sort(
        (a, b) => b.data.date.valueOf() - a.data.date.valueOf()
    );

    return rss({
        title: 'DZ | Technical Notes',
        description: 'Procedimientos operativos, SOPs y documentación técnica en SysAdmin & DevOps',
        site: context.site,

        items: sortedNotes.map((note) => {
            // Detectar idioma del path
            const lang = note.id.startsWith('es/') ? 'es' : 'en';
            const slug = note.id.replace(/^(es|en)\//, "");

            return {
                title: note.data.title,
                pubDate: note.data.date,
                description: note.data.description,
                link: `/${lang}/tech-notes/${slug}`,
                categories: [note.data.category, ...note.data.tags],
                author: note.data.author,
            };
        }),
        customData: `<language>es-es</language>`,
    });
}
```

PASO 7: Crear Layout Personalizado para Notas (Opcional)

Si quieres un diseño específico para las notas técnicas, crea:

`src/layouts/TechNoteLayout.astro`:

```
astro
```

```
--  
// src/layouts/TechNoteLayout.astro  
  
import type { CollectionEntry } from 'astro:content';  
import StarlightPage from '@/astrojs/starlight/components/StarlightPage.astro';  
  
interface Props {  
    entry: CollectionEntry<'tech-notes'>;  
}  
  
const { entry } = Astro.props;  
const { Content } = await entry.render();  
  
const formattedDate = entry.data.date.toLocaleDateString(entry.data.lang, {  
    year: 'numeric',  
    month: 'long',  
    day: 'numeric'  
});  
  
const updatedDate = entry.data.updated?.toLocaleDateString(entry.data.lang, {  
    year: 'numeric',  
    month: 'long',  
    day: 'numeric'  
});  
--  
  
<StarlightPage  
    frontmatter={ {  
        title: entry.data.title,  
        description: entry.data.description,  
    }}  
>  
<article class="tech-note">  
    <header class="note-header">  
        <div class="note-metadata">  
            <div class="badges">  
                <span class="category-badge">{entry.data.category}</span>  
                <span class="difficulty-badge">{entry.data.difficulty}</span>  
            </div>  
  
            <div class="dates">  
                <time datetime={entry.data.date.toISOString()}>  
                     {formattedDate}  
                </time>  
            </div>  
        </div>  
    </header>  
    <Content />  
</article>
```

```
{entry.data.updated && (
  <time datetime={entry.data.updated.toISOString()} class="updated">
    📅 Updated: {updatedDate}
  </time>
)
</div>

{entry.data.tags && entry.data.tags.length > 0 && (
  <div class="tags">
    {entry.data.tags.map(tag => (
      <span class="tag">#{tag}</span>
    )))
  </div>
)
</div>
</header>

<div class="note-content">
  <Content />
</div>

<footer class="note-footer">
  <p class="author">✍️ {entry.data.author}</p>
  {entry.data.systems && entry.data.systems.length > 0 && (
    <div class="systems">
      <span>💻 Systems:</span>
      {entry.data.systems.map(sys => (
        <span class="system-badge">{sys}</span>
      )))
    </div>
  )
  </footer>
</article>
</StarlightPage>

<style>
.tech-note {
  max-width: 800px;
  margin: 0 auto;
}

.note-header {
  margin-bottom: 2rem;
  padding-bottom: 1.5rem;
```

```
border-bottom: 2px solid var(--sl-color-gray-5);  
}
```

```
.note-metadata {  
  display: flex;  
  flex-direction: column;  
  gap: 1rem;  
}
```

```
.badges {  
  display: flex;  
  gap: 0.75rem;  
}
```

```
.category-badge,  
.difficulty-badge,  
.system-badge {  
  display: inline-block;  
  padding: 0.25rem 0.75rem;  
  border-radius: 4px;  
  font-size: 0.875rem;  
  font-weight: bold;  
  text-transform: uppercase;  
}
```

```
.category-badge {  
  background: var(--sl-color-accent);  
  color: var(--sl-color-black);  
}
```

```
.difficulty-badge {  
  background: var(--sl-color-gray-5);  
  color: var(--sl-color-white);  
}
```

```
.dates {  
  display: flex;  
  flex-direction: column;  
  gap: 0.25rem;  
  color: var(--sl-color-gray-3);  
  font-size: 0.875rem;  
}
```

```
.dates .updated {
```

```
    font-style: italic;
}

.tags {
  display: flex;
  flex-wrap: wrap;
  gap: 0.5rem;
}

.tag {
  background: var(--sl-color-gray-5);
  color: var(--sl-color-accent);
  padding: 0.25rem 0.75rem;
  border-radius: 4px;
  font-size: 0.875rem;
}

.note-content {
  line-height: 1.8;
}

.note-footer {
  margin-top: 3rem;
  padding-top: 1.5rem;
  border-top: 2px solid var(--sl-color-gray-5);
  display: flex;
  flex-direction: column;
  gap: 1rem;
}

.author {
  color: var(--sl-color-gray-3);
  font-style: italic;
}

.systems {
  display: flex;
  align-items: center;
  gap: 0.5rem;
  flex-wrap: wrap;
}

.system-badge {
  background: var(--sl-color-gray-6);
```

```
color: var(--sl-color-white);  
border: 1px solid var(--sl-color-gray-5);  
}  
</style>
```

Estructura de Archivos Final

```
daniel-zamo.github.io/  
├── astro.config.mjs      ( ✅ Actualizado)  
├── src/  
│   ├── content/  
│   │   ├── config.ts      ( ✅ Actualizado)  
│   │   ├── docs/          (Documentación principal)  
│   │   │   ├── en/  
│   │   │   │   ├── az-cloud-ops/  
│   │   │   │   ├── linux-ops/  
│   │   │   │   ├── projects/  
│   │   │   │   └── index.mdx  
│   │   └── es/  
│   │       ├── az-cloud-ops/  
│   │       ├── linux-ops/  
│   │       ├── projects/  
│   │       └── index.mdx  
│   ├── tech-notes/        ( ✅ NUEVO)  
│   │   ├── en/  
│   │   │   ├── sop/  
│   │   │   ├── troubleshooting/  
│   │   │   └── howto/  
│   │   └── es/  
│   │       ├── sop/  
│   │       ├── troubleshooting/  
│   │       └── howto/  
│   └── questions/  
└── layouts/  
    └── TechNoteLayout.astro  ( ✅ NUEVO - Opcional)  
└── pages/  
    ├── rss.xml.js         ( ✅ Actualizado)  
    └── tech-notes/  
        └── index.astro      ( ✅ NUEVO)  
└── styles/  
    └── custom.css
```

```
|--- public/  
|   package.json
```

Casos de Uso y Ejemplos

Ejemplo 1: Crear una Nueva Nota Técnica SOP

Archivo: `[src/content/tech-notes/es/sop/resolucion-dns-interno.md]`

markdown

title: "Resolución DNS en Red Interna con systemd-resolved"

description: "Procedimiento operativo estándar para configurar resolución DNS personalizada en entornos Linux con systemd"

date: 2026-01-31

tags: ['dns', 'linux', 'systemd', 'networking']

category: 'sop'

systems: ['linux']

difficulty: 'intermediate'

author: 'Daniel Zamo'

lang: 'es'

keywords: ['dns', 'systemd-resolved', 'resolv.conf', 'linux networking']

🎯 Objetivo

Configurar resolución DNS personalizada en un servidor Linux utilizando `systemd-resolved` para apuntar a servidores DNS

📄 Prerrequisitos

- Sistema Linux con systemd (Ubuntu 20.04+, RHEL 8+, Debian 10+)
- Acceso root o sudo
- Direcciones IP de los servidores DNS internos

🔧 Procedimiento

1. Verificar Estado de systemd-resolved

```
```bash
systemctl status systemd-resolved
````
```

2. Editar Configuración Principal

```
```bash
sudo nano /etc/systemd/resolved.conf
````
```

Descomentar y configurar:

```
```ini
[Resolve]
DNS=10.0.10.53 10.0.10.54
FallbackDNS=8.8.8.8 1.1.1.1
Domains=~empresa.local
DNSSEC=allow-downgrade
````
```

3. Aplicar Cambios

```
```bash
sudo systemctl restart systemd-resolved
sudo resolvectl status
````
```

✅ Validación

```
```bash
Verificar servidores DNS activos
resolvectl query empresa.local
```

# Comprobar resolución

```
dig @10.0.10.53 intranet.empresa.local
````
```

🚨 Troubleshooting

****Problema**:** DNS no resuelve después del cambio

****Solución**:** Verificar que `/etc/resolv.conf` sea un symlink a `/run/systemd/resolve/stub-resolv.conf`

```
```bash
ls -la /etc/resolv.conf
````
```

📖 Referencias

- [Documentación systemd-resolved](<https://www.freedesktop.org/software/systemd/man/systemd-resolved.service.html>)
- RFC 1035 (DNS)

Ejemplo 2: Nota de Troubleshooting

Archivo: [\(src/content/tech-notes/es/troubleshooting/ssh-connection-timeout.md\)](#)

markdown

title: "Resolución: Timeout en Conexiones SSH después de Actualización"
description: "Documentación de incidente resuelto: conexiones SSH fallando con timeout tras actualización de sistema operativo"
date: 2026-01-28
updated: 2026-01-31
tags: ['ssh', 'troubleshooting', 'firewall', 'selinux']
category: 'troubleshooting'
systems: ['linux']
difficulty: 'advanced'
author: 'Daniel Zamo'
lang: 'es'

🚨 Descripción del Incidente

Fecha: 28 de enero de 2026

Sistema Afectado: Servidores de producción RHEL 9.2

Síntoma: Conexiones SSH desde red corporativa fallan con timeout después de actualización del kernel

🔎 Diagnóstico

1. Verificación de Conectividad de Red

```
```bash
Ping OK
ping -c 4 server-prod-01
```

# Telnet a puerto 22 falla

```
telnet server-prod-01 22
Error: Connection timed out
````
```

2. Revisión de Logs del Servidor

```
```bash
sudo journalctl -u sshd -n 50 --no-pager
```

# No se observan intentos de conexión registrados
````

3. Análisis de Firewall

```
```bash
sudo firewall-cmd --list-all
```

# Salida:

```
public (active)
 interfaces: ens160
 services: dhcipv6-client
X SSH NO está en la lista de servicios permitidos
```

```

Resolución

Causa Raíz

La actualización del sistema operativo resetó la configuración de firewalld a valores por defecto, eliminando la regla que perm

Solución Aplicada

```
```bash
Permitir SSH permanentemente
sudo firewall-cmd --permanent --add-service=ssh
```

# Recargar configuración

```
sudo firewall-cmd --reload
```

# Verificar

```
sudo firewall-cmd --list-services
Salida esperada: dhcipv6-client ssh
```

```

Impacto

- **Duración**: 45 minutos
- **Usuarios Afectados**: 12 administradores de sistemas
- **Servicio**: Conexión SSH (puerto 22)

Prevención Futura

1. Implementar configuración de firewall gestionada por Ansible
2. Añadir verificación de servicios críticos en checklist post-actualización
3. Configurar monitoreo de accesibilidad SSH desde jumphost

Lecciones Aprendidas

- Siempre verificar configuración de firewall tras actualizaciones de sistema operativo
- Documentar reglas de firewall en repositorio de configuración
- Mantener acceso de emergencia mediante consola serial/BMC

Configuración Avanzada

Filtrado por Tags

Crea `src/pages/tech-notes/tags/[tag].astro` para páginas de tags:

```
astro
```

```
---
```

```
import { getCollection } from 'astro:content';
import StarlightPage from '@astrojs/starlight/components/StarlightPage.astro';

export async function getStaticPaths() {
  const allNotes = await getCollection('tech-notes');
  const uniqueTags = [...new Set(allNotes.flatMap(note => note.data.tags))];

  return uniqueTags.map(tag => ({
    params: { tag },
    props: { tag }
  }));
}

const { tag } = Astro.props;
const notes = await getCollection('tech-notes', ({ data }) => {
  return data.tags.includes(tag);
});
---

<StarlightPage
  frontmatter={{
    title: `Tag: ${tag}`,
    description: `Technical notes tagged with '${tag}'`
  }}
>
<h1>Notes tagged with: <strong>#${tag}</strong></h1>

<ul>
  {notes.map(note => (
    <li>
      <a href={`/tech-notes/${note.id}`}>{note.data.title}</a>
    </li>
  )));
</ul>
</StarlightPage>
```

Verificación Final

Comandos de Testing

```
bash

# 1. Verificar que no hay errores de sintaxis
npm run build

# 2. Previsualizar el sitio localmente
npm run dev

# 3. Verificar las rutas generadas
# - http://localhost:4321/es/tech-notes/
# - http://localhost:4321/en/tech-notes/
# - http://localhost:4321/es/tech-notes/sop/nombre-de-nota/
```

Checklist de Validación

- Las notas aparecen en el índice `/tech-notes/`
 - El sidebar muestra el enlace "Notas Técnicas"
 - El RSS feed funciona en `/rss.xml`
 - Las rutas respetan el idioma (es/en)
 - Los metadatos (fecha, tags) se muestran correctamente
 - El sitio compila sin errores (`npm run build`)
-

Ventajas de esta Implementación

- Separación de Conceptos:** Documentación técnica vs. Notas operativas
 - Schema Específico:** Metadatos diseñados para SOPs y troubleshooting
 - Sin Dependencias Externas:** No requiere plugins de terceros
 - Multiidioma Nativo:** Soporte completo para es/en
 - Escalable:** Fácil añadir categorías y filtros
 - SEO Optimizado:** Metadata rica para buscadores
 - Feed RSS:** Suscripción automática
-

Próximos Pasos Opcionales

1. **Sistema de Búsqueda:** Integrar Pagefind o Algolia
2. **Versionado de SOPs:** Añadir campo `version` al schema

3. **Comentarios:** Integrar Giscus o similar
 4. **Analytics:** Rastrear qué procedimientos son más consultados
 5. **Export a PDF:** Generar PDFs de SOPs automáticamente
-

Autor: Guía generada por Claude (Anthropic)

Fecha: 31 de enero de 2026

Versión: 1.0