

Configuración de Email en JavaScript/Node.js

Para Aumentar Entrega en Inbox (Evitar Spam)

Para: Red Team Xal Digital

Dominio: xaldigital.com

Email de prueba: recursos.humanos@xaldigital.com

Objetivo

Configurar correctamente el envío de emails desde JavaScript/Node.js para que:

- Pasen autenticación SPF/DKIM/DMARC - Lleguen a Inbox (no a Spam) -
 - Se vean como emails legítimos
-

Opción 1: Usando Nodemailer (Recomendado)

Instalación

```
npm install nodemailer
```

Código Base

```
const nodemailer = require('nodemailer');

// Configurar transporter
const transporter = nodemailer.createTransport({
    host: 'mail.xaldigital.com',           // Tu servidor SMTP
    port: 465,                            // Puerto SSL
    secure: true,                         // Usa SSL/TLS
    auth: {
        user: 'recursos.humanos@xaldigital.com',
        pass: 'TU CONTRASEÑA AQUI'        // Usa variable de entorno
    }
});

// Opciones del email
const mailOptions = {
    from: 'Recursos Humanos <recursos.humanos@xaldigital.com>',
    to: 'empleado@xaldigital.com',
    subject: 'Documento compartido: Reporte Confidencial Q4',
    html: htmlContent,

    // Headers importantes para autenticación
    headers: {
```

```

        'X-Mailer': 'Corporate-Mail-System/1.0',
        'X-Priority': '3',
        'X-MSMail-Priority': 'Normal',
        'Content-Type': 'text/html; charset=utf-8',
        'MIME-Version': '1.0'
    }
};

// Enviar email
transporter.sendMail(mailOptions, (error, info) => {
    if (error) {
        console.log(' Error:', error);
    } else {
        console.log(' Email enviado:', info.response);
    }
});

```

Opción 2: Headers Avanzados (Para máxima autenticación)

```

const nodemailer = require('nodemailer');

const transporter = nodemailer.createTransport({
    host: 'mail.xaldigital.com',
    port: 465,
    secure: true,
    auth: {
        user: 'recursos.humanos@xaldigital.com',
        pass: process.env.SMTP_PASSWORD // + Usar variable de entorno
    },
    // Opciones avanzadas
    tls: {
        rejectUnauthorized: false // Para testing (no en producción)
    }
});

const mailOptions = {
    from: {
        name: 'Recursos Humanos',
        address: 'recursos.humanos@xaldigital.com'
    },
    to: recipient,
    subject: 'Documento importante compartido',
}

```

```

    html: htmlContent,

    // Headers para pasar autenticación
    headers: {
        'X-Mailer': 'Corporate-Mail-System/1.0',
        'X-Priority': '3',
        'X-MSMail-Priority': 'Normal',
        'Importance': 'High',
        'Content-Type': 'text/html; charset=utf-8',
        'MIME-Version': '1.0',
        'X-Originating-IP': '[192.168.1.1]',
        'X-Mailer-Version': '1.0'
    }
};

await transporter.sendMail(mailOptions);

```

Opción 3: Usando SendGrid (Si no tienes servidor SMTP)

Instalación

```
npm install @sendgrid/mail
```

Código

```

const sgMail = require('@sendgrid/mail');
sgMail.setApiKey(process.env.SENDGRID_API_KEY);

const msg = {
    to: 'empleado@xaldigital.com',
    from: 'recursos.humanos@xaldigital.com', // ← Debe estar verificado en SendGrid
    subject: 'Documento compartido',
    html: htmlContent,
    replyTo: 'recursos.humanos@xaldigital.com',
    headers: {
        'X-Priority': '3',
        'X-Mailer': 'Corporate-Mail-System/1.0'
    }
};

await sgMail.send(msg);

```

Plantilla HTML Optimizada

```
const htmlContent = `
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width">
    <style>
        body { font-family: Arial, sans-serif; }
        .header { background: #4285f4; color: white; padding: 20px; }
        .content { padding: 30px; }
        .button {
            background: #4285f4;
            color: white;
            padding: 12px 30px;
            text-decoration: none;
            display: inline-block;
            border-radius: 4px;
        }
    </style>
</head>
<body>
    <div class="header">
        <h2>Google Drive</h2>
    </div>

    <div class="content">
        <p>Hola Empleado,</p>
        <p>Recursos Humanos ha compartido un documento importante contigo en Google Drive:</p>
        <div style="background: #f5f5f5; padding: 20px; margin: 20px 0; border-radius: 8px;">
            <p style="font-weight: bold;"> Reporte_Confidencial_Q4_2024.pdf</p>
            <p style="color: #999; font-size: 12px;">Compartido hace 5 minutos</p>
        </div>

        <div style="text-align: center; margin: 30px 0;">
            <a href="https://xaldigital.com/verify.php" class="button">Abrir Documento</a>
        </div>

        <p style="color: #999; font-size: 12px; margin-top: 30px;">
            Este correo contiene información confidencial.
            Si no eras el destinatario, por favor elimínalo.
        </p>
    </div>

```

```
</body>
</html>
`;
```

Test de Envío

Script de Prueba Completo

```
const nodemailer = require('nodemailer');

async function testEmail() {
    const transporter = nodemailer.createTransport({
        host: 'mail.xaldigital.com',
        port: 465,
        secure: true,
        auth: {
            user: 'recursos.humanos@xaldigital.com',
            pass: process.env.SMTP_PASSWORD
        }
    });

    // Generar dirección de mail-tester
    console.log('1. Ve a https://www.mail-tester.com/');
    console.log('2. Copia la dirección de prueba (ej: test-xxxxxx@mail-tester.com)');
    console.log('3. Modifica "testEmail" abajo');

    const testEmail = 'test-xxxxxx@mail-tester.com'; // ← CAMBIA ESTO

    const mailOptions = {
        from: 'Recursos Humanos <recursos.humanos@xaldigital.com>',
        to: testEmail,
        subject: 'Correo de Prueba - Red Team Xal',
        html: '<p>Este es un correo de prueba para verificar autenticación SPF/DKIM/DMARC</p>',
        headers: {
            'X-Mailer': 'Corporate-Mail-System/1.0',
            'X-Priority': '3'
        }
    };

    try {
        const info = await transporter.sendMail(mailOptions);
        console.log(' Email enviado exitosamente');
        console.log(' ID:', info.messageId);
        console.log('\n4. Regresa a https://www.mail-tester.com/ y haz clic en "Check your s');
    } catch (error) {
```

```
        console.error(' Error al enviar:', error.message);
    }
}

testEmail();
```

Ejecutar Test

```
# Linux/Mac
SMTP_PASSWORD="tu_contraseña" node test-email.js

# Windows (PowerShell)
$env:SMTP_PASSWORD="tu_contraseña"; node test-email.js

# O usar archivo .env
# npm install dotenv
```

Variables de Entorno (Seguridad)

Archivo .env

```
SMTP_HOST=mail.xaldigital.com
SMTP_PORT=465
SMTP_USER=recursos.humanos@xaldigital.com
SMTP_PASSWORD=tu_contraseña_aqui
SMTP_SECURE=true
```

Cargar en Node.js

```
require('dotenv').config();

const transporter = nodemailer.createTransport({
  host: process.env.SMTP_HOST,
  port: process.env.SMTP_PORT,
  secure: process.env.SMTP_SECURE === 'true',
  auth: {
    user: process.env.SMTP_USER,
    pass: process.env.SMTP_PASSWORD
  }
});
```

Ejemplo Completo: Campaña de 250 Empleados

```
const nodemailer = require('nodemailer');
require('dotenv').config();

// Lista de empleados
const empleados = [
  { email: 'juan.lopez@xaldigital.com', nombre: 'Juan López' },
  { email: 'maria.garcia@xaldigital.com', nombre: 'María García' },
  // ... más empleados
];

// Configurar transporter
const transporter = nodemailer.createTransport({
  host: process.env.SMTP_HOST,
  port: 465,
  secure: true,
  auth: {
    user: process.env.SMTP_USER,
    pass: process.env.SMTP_PASSWORD
  }
});

// Template HTML
const getHtmlTemplate = (nombre) => `
<div style="font-family: Arial, sans-serif;">
  <p>Hola ${nombre},</p>
  <p>Se ha compartido un documento importante contigo.</p>
  <a href="https://xaldigital.com/verify.php?email=${encodeURIComponent(empleados.find(e =>
    e.nombre === nombre).email)}>
    Abrir Documento
  </a>
</div>
`;

// Función para enviar emails espaciados
async function enviarCampaña() {
  let enviados = 0;
  let errores = 0;

  for (const empleado of empleados) {
    try {
      const mailOptions = {
        from: 'Recursos Humanos <recursos.humanos@xaldigital.com>',
        to: empleado.email,
        subject: 'Documento compartido en Google Drive',
        html: getHtmlTemplate(empleado.nombre),
      };
      await transporter.sendMail(mailOptions);
      enviados++;
    } catch (error) {
      errores++;
    }
  }
}
```

```

        headers: {
          'X-Mailer': 'Corporate-Mail-System/1.0',
          'X-Priority': '3'
        }
      };

      await transporter.sendMail(mailOptions);
      enviados++;
      console.log(` [${enviados}] Enviado a: ${empleado.email}`);
    }

    // Esperar 2 segundos entre emails (no saturar servidor)
    await new Promise(resolve => setTimeout(resolve, 2000));

  } catch (error) {
    errores++;
    console.error(` Error enviando a ${empleado.email}: `, error.message);
  }
}

console.log(`\n Resumen:`);
console.log(` Enviados: ${enviados}`);
console.log(` Errores: ${errores}`);
}

enviarCampaña();

```

Verificar Autenticación

Endpoint para Testing

```

const express = require('express');
const app = express();

app.get('/test-smtp', async (req, res) => {
  try {
    await transporter.verify();
    res.json({
      status: 'success',
      message: 'SMTP conectado correctamente'
    });
  } catch (error) {
    res.json({
      status: 'error',
      message: error.message
    });
  }
}

```

```
    }
});

app.listen(3000);
```

Debugging

Ver detalles de envío

```
const transporter = nodemailer.createTransport({
    // ... config
});

// Habilitar debug
transporter.set('logger', true);
transporter.set('debug', true);

// Ver logs detallados
transporter.sendMail(mailOptions, (error, info) => {
    if (error) {
        console.log('Headers:', error);
    }
});
```

Checklist JavaScript

- Instalé nodemailer: `npm install nodemailer`
 - Configuré credenciales SMTP correctamente
 - Usé headers avanzados
 - Probé con mail-tester.com
 - Score mail-tester 8/10
 - Envié correo de prueba a Gmail
 - Verificué que llegó a Inbox (no Spam)
 - Implementé delay entre envíos (2 seg)
 - Registré logs de envío
-

Errores Comunes

Error: “Autenticación fallida”

```
// CORRECTO
user: 'recursos.humanos@xaldigital.com',
```

```
pass: 'contraseña_exacta'

// INCORRECTO
user: 'recursos.humanos@xaldigital', // Falta .com
pass: 'contraseña (con espacio)' // Espacios extras
```

Error: “Conexión rechazada”

```
// Verifica que mail.xaldigital.com esté activo
// Usa puerto 465 (SSL) o 587 (TLS)
secure: true, // Para puerto 465
port: 465,
// O
secure: false, // Para puerto 587
port: 587,
```

Error: “Correo en spam”

- Verifica SPF/DKIM/DMARC configurados (paso anterior)
 - Espera 30 minutos a que se propague DNS
 - Prueba con mail-tester.com
-

Preguntas Frecuentes

¿Necesito mail.xaldigital.com?

R: No, necesitas el servidor SMTP de tu hosting. Pregunta a tu proveedor.

¿Puedo cambiar el remitente?

R: Sí, pero debe ser un email válido configurado en tu hosting.

¿Cuántos emails puedo enviar?

R: Depende de tu hosting. Típicamente 100-1000 por hora.

¿Se ve el remitente real en Gmail?

R: Gmail mostrará Recursos Humanos <recursos.humanos@xaldigital.com>

Resultado Esperado

Después de configurar correctamente:

Email enviado desde: recursos.humanos@xaldigital.com
Llega a: Inbox (no Spam)
Gmail muestra: Autenticado
Mail-tester score: 9-10 de 10

¡Lista para la campaña de 250+ empleados!