

PROTOCOLO REALNODE

Guía de Integración B2B — Vanilla JS

Manual técnico detallado de implementación.

Arquitectura Universal

Este documento detalla la integración técnica del protocolo RealNode en su plataforma utilizando Vanilla JS. La interfaz que integrará está diseñada para adaptarse dinámicamente a su nivel de servicio (**RN Insight**, **RN Sentinel** o **RN Vault**) sin necesidad de modificar el código posteriormente.

1 Paso 0: Requisitos y Claves API

Antes de comenzar a programar, debe obtener sus accesos a la infraestructura de RealNode.

1. **Creación de cuenta:** Diríjase al portal de administración de RealNode (<https://app.realnode.emkaylabs.tech/signup>) para crear su cuenta empresarial. El formulario de registro rápido solo solicitará: **Nombre de la Empresa**, **Correo Profesional** y una **Contraseña de Administrador**.
2. **Generación de claves:** Al finalizar este registro, el portal generará un **par de claves**: una **Clave Pública** (ej: `pk_live_XXX`) destinada a su sitio web, y una **Clave Secreta** (ej: `sk_live_XXX`) reservada estrictamente para su servidor backend.
3. **Conservación:** Estas claves identifican y aseguran sus solicitudes. Mantenga su Clave Secreta en absoluta seguridad dentro de su archivo `.env`. Nunca la exponga en el navegador.
4. **Nota sobre el plan elegido:** El campo `event_id` en la API `/consume` depende del plan suscrito:
 - **RN Insight** : no se requiere `event_id` — solo análisis pasivo.
 - **RN Sentinel** : `event_id` opcional — consumo por sesión.
 - **RN Vault** : `event_id` **obligatorio** — cree el evento desde su panel de administración antes de llamar a `/consume`.

2 Paso 1: Inicialización del SDK

El primer paso técnico consiste en cargar el motor de RealNode en su sitio.

Ubicación recomendada: Este bloque debe insertarse **en todas las páginas** que requieran la seguridad de RealNode (página de pago, área de cliente), inmediatamente antes de la etiqueta de cierre `</body>`.

```
<script>
// Configuración global para inicializar RealNode
window.RN_CONFIG = {
  // Utilice NICAMENTE su Clave Pública (comienza con pk_)
  apiKey: "pk_live_SU-CLAVE-PUBLICA-AQUI",
  clientId: "nombre-de-su-tienda" // Opcional, til para filtrar logs
}
```

```
};  
</script>  
<!-- Carga del motor de seguridad asncrono -->  
<script type="module" src="https://api.emkaylabs.tech/rn-client.js"></script>
```

3 Paso 2: Integración de la Interfaz de Usuario (UI)

Ahora debe integrar los elementos visuales necesarios para la interacción del usuario. El código está estructurado en módulos lógicos.

3.1. 2.1. El estilo global (CSS)

Ubicación recomendada: Integrar en su hoja de estilos principal (ej: `styles.css`), o en la etiqueta `<head>` de sus páginas.

```
<style>
.rn-panel { background: #111; color: #fff; padding: 25px; border-radius:
    12px; font-family: sans-serif; }
.rn-badge { background: rgba(0, 255, 136, 0.1); color: #00ff88; padding: 5
    px 10px; border-radius: 4px; }
.rn-btn { background: #00ff88; color: #000; border: none; padding: 10px 15
    px; border-radius: 6px; cursor: pointer; font-weight: bold; margin-
    right: 5px; }
.rn-btn-secondary { background: #444; color: #fff; border: none; padding:
    10px 15px; border-radius: 6px; cursor: pointer; }
.rn-btn-danger { background: rgba(239, 68, 68, 0.1); color: #ef4444;
    border: 1px solid #ef4444; }
.rn-input { background: #222; border: 1px solid #444; color: #fff; padding
    : 10px; border-radius: 6px; width: 100%; margin-bottom: 10px; }
.rn-device-row { display: flex; justify-content: space-between; align-
    items: center; border-bottom: 1px solid #222; padding: 15px 0; }
</style>
```

3.2. 2.2. La Zona de Pago

Ubicación recomendada: Integrar en el DOM de su **página de confirmación de pedido**, en el lugar previsto para el botón de acción principal.

```
<div style="margin-bottom: 30px;">
  <p>Entradas autorizadas: <span id="rn-quota-user" class="rn-badge">--<
    /span> / <span id="rn-quota-max">--</span></p>
  <input type="number" id="rn-ticket-quantity" class="rn-input" value="1
    " min="1" style="width: 80px; display: inline-block;">
  <button id="rn-btn-acheter" data-rn-protect class="rn-btn">Comprar de
    forma segura</button>
  <button id="rn-btn-demander-quota" style="display:none;" class="rn-btn
    ">Solicitar m s entradas</button>
</div>
```

3.3. 2.3. La Zona de Perfil de Usuario (Gestión de dispositivos)

Ubicación recomendada: Integrar en el panel de usuario (“Mi Cuenta” o “Perfil”).

```
<div class="rn-panel">
  <h3>Mis Dispositivos Seguros</h3>
  <button id="rn-btn-revente" class="rn-btn-secondary">Liberar mi
    dispositivo (Para reventa)</button>
  <div id="rn-devices-list" style="margin-top:15px;">Cargando...</div>

  <h3 style="margin-top: 30px;">Transferir mis accesos a un tercero</h3>
  <input type="text" id="rn-transfer-id" class="rn-input" placeholder="
    ID del destinatario">
  <button id="rn-btn-transferer" class="rn-btn-secondary">Transferir mi
    cuenta</button>
```

```
<h3 style="margin-top: 30px;">Restauraci n de Emergencia</h3>
<input type="text" id="rn-recovery-code" class="rn-input" placeholder=
  "C digo de Recuperaci n (Ej: RN-REC-...)">
<button id="rn-btn-recuperer" class="rn-btn">Restaurar mediante
  C digo</button>

<p style="margin-top:10px; font-size: 0.85em; color: #aaa;"> Perdi
  su c digo?</p>
<input type="email" id="rn-rescue-email" class="rn-input" placeholder=
  "su@email.com">
<button id="rn-btn-rescue" class="rn-btn-secondary">Solicitar enlace
  de Recuperaci n</button>
</div>
```

4 Paso 3: Lógica Frontend (JavaScript)

Este módulo JavaScript organiza la integración en el lado del cliente. Asegura la conexión entre la interfaz de usuario y la API criptográfica de RealNode.

Ubicación recomendada: Incluir en su archivo JavaScript principal (ej: main.js) o al final del documento mediante una etiqueta `<script>`.

```
document.addEventListener("DOMContentLoaded", async () => {

    // 0. MOSTRAR CUOTAS EN TIEMPO REAL
    // El evento 'rn-purchase-success' se emite tras cada verificación exitosa
    window.addEventListener('rn-purchase-success', (e) => {
        const result = e.detail;
        if (result.remaining !== undefined) {
            document.getElementById('rn-quota-user').innerText = result.remaining;
            document.getElementById('rn-quota-max').innerText = result.max_limit;
            if (result.remaining === 0) {
                document.getElementById('rn-btn-acheter').style.display = 'none';
                document.getElementById('rn-btn-demander-quota').style.display = 'inline-block';
            }
        }
    });

    // Restauración de sesión (si el usuario volvió a la página)
    try {
        await window.RN_Monitor.restoreSession();
    } catch(e) { /* El usuario aún no tiene una sesión activa */ }

    // 1. CARGA Y GESTIÓN DE DISPOSITIVOS (Planes V2/V3)
    const listDiv = document.getElementById('rn-devices-list');
    setTimeout(() => {
        if (listDiv.innerHTML === 'Cargando...') listDiv.innerHTML = 'Dispositivos no rastreados (Plan Insight)';
    }, 3000);

    // El evento 'rn:v2:sensor-ready' se emite cuando el perfil de usuario está listo
    window.addEventListener('rn:v2:sensor-ready', async () => {
        try {
            const devices = await window.RN_Monitor.listDevices(window.RN_Monitor.currentIdh);
            listDiv.innerHTML = '';
            devices.data.forEach(dev => {
                listDiv.innerHTML += `
                <div class="rn-device-row">
                    <span style="font-family: monospace;">Dispositivo : ${dev.device_hash.substring(0,8)}...</span>
                    <button onclick="revocDevice('${dev.device_hash}')" class="rn-btn rn-btn-danger">Bloquear (Robo)</button>
                </div>`;
            });
        } catch (e) { /* Ignorado para el nivel RN Insight (pasivo) */ }
    });
});
```

```

// Funciones de utilidad para la gesti n de perfiles
window.revocDevice = async (hash) => {
  if(confirm(" Bloquear este dispositivo en la red?")) {
    await window.RN_Monitor.revokeDevice(hash);
    alert("Dispositivo bloqueado con éxito.");
  }
};

document.getElementById('rn-btn-revente').onclick = async () => {
  if(confirm(" Limpiar este dispositivo de su identidad antes de
  revenderlo?")) {
    await window.RN_Monitor.requestResale(window.RN_Monitor.
    currentIdh);
    alert("Dispositivo liberado.");
  }
};

document.getElementById('rn-btn-transferer').onclick = async () => {
  const newId = document.getElementById('rn-transfer-id').value;
  if(confirm(" Transferir irreversiblemente su cuenta a este
  usuario?")) {
    try { await window.RN_Monitor.transferDevice(newId); alert("
    Transferencia exitosa."); }
    catch (e) { alert("Error durante la transferencia."); }
  }
};

// 3. LA L GICA DE COMPRA (N CLEO DE LA INTEGRACI N)
// NOTA: data-rn-protect asegura que el SDK intercepte el clic
// autom ticamente. El evento 'rn:verified' se emite tras la
// biometr a.
document.getElementById('rn-btn-acheter').addEventListener('rn:
verified', (e) => {
  const result = e.detail;
  const desiredQuantity = parseInt(document.getElementById('rn-
ticket-quantity').value) || 1;

  if (result.remaining !== undefined) {
    document.getElementById('rn-quota-user').innerText = result.
remaining;
    document.getElementById('rn-quota-max').innerText = result.
max_limit;
  }

  if (result.status === 'authorized') {
    if (result.remaining === undefined || desiredQuantity <=
result.remaining) {
      const rnPayload = {
        quantity: desiredQuantity,
        idh: window.RN_Monitor.currentIdh,
        device_hash: result.device_hash,
        device_token: localStorage.getItem("rn_device_token"),
        client_id: window.RN_CONFIG.clientId || "rn-universal-
client"
      };

      // -> ENV O A SU SERVIDOR <-
      fetch('/api/su-checkout', {
        method: 'POST',
        headers: { 'Content-Type': 'application/json' },

```

```

        body: JSON.stringify(rnPayload)
    }).then(response => {
        alert("Proceso de seguridad validado. Finalizando la
            transacci n...");
    });
    } else { alert("Operaci n rechazada: Cuota insuficiente."); }
}
});

// 4. RESTAURACI N Y EXTENSI N
document.getElementById('rn-btn-demander-quota').onclick = async () =>
{
    await window.RN_Monitor.requestQuotaExtension(); alert("Solicitud
        enviada.");
};
document.getElementById('rn-btn-recuperer').onclick = async () => {
    const code = document.getElementById('rn-recovery-code').value;
    try { await window.RN_Monitor.recover(code); alert("Identidad
        restaurada!"); }
    catch (error) { alert("C digo de recuperaci n no v lido."); }
};
document.getElementById('rn-btn-rescue').onclick = async () => {
    const email = document.getElementById('rn-rescue-email').value;
    try { await window.RN_Monitor.requestRescue(email); alert("Enlace
        enviado por correo electr nico."); }
    catch (error) { alert("Error durante la solicitud."); }
};
});

```

5 Paso 4: Validación de la Transacción (Su Backend)

Una vez que su servidor (ej: Node.js, PHP, Python) ha recibido el `rnPayload` desde su frontend, debe transmitir estas pruebas a la API de RealNode.

Implementación: Debe crear un punto final en su servidor (ej: `/api/checkout`) capaz de recibir el `rnPayload` del frontend, y luego ejecutar la solicitud HTTP de atestación descrita a continuación hacia la infraestructura de RealNode.

Seguridad Crítica (Doble Clave) : Para este paso de servidor a servidor, debe utilizar su **Clave Secreta** (`sk_live_...`) obtenida en su área de administrador. Nunca coloque esta clave secreta en su código JavaScript del frontend.

Solicitud a realizar desde su backend a RealNode:

```
POST https://api.emkaylabs.tech/consume
Headers:
  Content-Type: application/json
  X-RealNode-API-Key: pk_live_SU-CLAVE-PUBLICA
  X-RealNode-Secret-Key: sk_live_SU-CLAVE-SECRETA
Body:
{
  "quantity": 3,
  "idh": "ID-RECUPERADO-EN-EL-FRONTEND",
  "device_hash": "RN-DEVICE-XXXX",
  "device_token": "TOKEN-XXX",
  "client_id": "nombre-de-su-tienda",
  "event_id": "EVT-XXXX" // Obligatorio para RN Vault
}
```

Nota: el campo `event_id` solo es obligatorio para el plan RN Vault. Corresponde al identificador del evento creado en su panel de administración.

6 Paso 5: Pruebe su Integración

Para asegurarse de que todo está configurado correctamente, realice estas dos pruebas simples:

1. **Prueba Frontend (Cuotas) :** Abra su página de pago en su navegador. Si la integración es exitosa, verá el contador “Entradas autorizadas” cambiar de - a su cuota real (ej: 5 / 5) unos segundos después de cargar la página.
2. **Prueba Backend (Consumo) :** Desde el terminal de su servidor, puede simular una validación del backend con este comando `curl`:

```
curl -X POST https://api.emkaylabs.tech/consume \
-H "Content-Type: application/json" \
-H "X-RealNode-API-Key: pk_live_SU-CLAVE-PUBLICA" \
-H "X-RealNode-Secret-Key: sk_live_SU-CLAVE-SECRETA" \
-d '{
  "quantity": 1,
  "idh": "test-id",
  "device_hash": "test-hash",
  "device_token": "test-token",
  "client_id": "test-client"
  // Agregar "event_id": "EVT-XXXX" si plan RN Vault
}'
```

Respuesta HTTP 200 : comunicación servidor a servidor operativa.
Respuesta HTTP 403 : verifique sus claves API.
Error “Invalid event ID” : añada el campo `event_id` (solo para plan RN Vault).

7 Paso 6: Casos Especiales y Buenas Prácticas

7.1. 6.1. Integración en React / Vue / Angular (SPA)

Los ejemplos del Paso 3 utilizan `DOMContentLoaded`. En una Single Page Application (SPA), los componentes se montan dinámicamente.

- **React** : Coloque la lógica de JavaScript en un hook `useEffect()` en lugar de `DOMContentLoaded`.
- **Vue.js** : Coloque la lógica en el ciclo de vida `mounted()`.
- **Angular** : Utilice `ngOnInit()`.

Asegúrese de limpiar sus oyentes de eventos (`removeEventListener`) al desmontar el componente para evitar fugas de memoria.

7.2. 6.2. Content Security Policy (CSP)

Si su sitio utiliza encabezados CSP estrictos, el script de RealNode puede ser bloqueado. Autorice nuestro dominio en la configuración de su servidor:

```
Content-Security-Policy:  
script-src 'self' https://api.emkaylabs.tech https://*.hanko.io;  
connect-src 'self' https://api.emkaylabs.tech https://*.hanko.io;
```

Soporte Técnico : Para cualquier duda de integración, contacte con nuestro equipo en **security@emkaylabs.tech**. Respondemos en menos de 24 horas.