GS515 – Digital Signatures and Key Vault Integration in Globalization Studio

Digital signatures are a key part of global compliance. Whether you're submitting invoice XMLs to a government platform or protecting PDF layouts from tampering, a digital signature proves:

- ✅ The document was sent by your company
- ✅ The content hasn't been changed after generation
- ✅ The file meets government or customer legal requirements

In this article, we'll explore:

- What a digital signature is and how it works
- How signing is handled in **Electronic Invoicing** features
- How it differs from digital handling in **Spain's SII (MTD)**
- How to configure **Azure Key Vault** for digital signing in Dynamics 365
- Tips for managing certificates across regions

📘 Builds on previous articles:
GS506 – Pipelines, GS510 – Feature Reuse, and GS514 – Document Routing

---

🔐 What Is a Digital Signature?

A **digital signature** is a cryptographic seal applied to an electronic document to prove its authenticity and integrity. It's generated using a certificate file (usually .pfx) issued by a trusted certification authority like FNMT, Camerfirma, or GlobalSign.

Once a document is signed:

- It is cryptographically locked
- Any tampering or changes invalidate the signature
- Government or business receivers can verify the sender identity

---

🧩 Where Digital Signatures Are Used in Globalization Studio

Scenario	Framework	Signing Location	Signature Required?
Spain – Facturae (e-invoice XML)	Globalization Studio	Inside XML payload	✅ Required
Italy – FatturaPA	Globalization Studio	Embedded XML signature	✅ Required
Saudi Arabia – Phase 2 e-invoice	Globalization Studio	XML/QR Code	✅ Required
France – Invoice compliance (optional)	Globalization Studio	Embedded XML or PDF signature	Optional
Spain – SII / MTD	Electronic Messaging	Applied at HTTP transport level	✅ Required

---

🧠 In Globalization Studio, digital signatures are configured in **pipeline steps** using certificates stored in **Azure Key Vault**.

📌 For Spain's **SII (MTD)**, signing is handled at the **transport level** using the Electronic Messaging framework , not in the XML content, and not via Globalization Studio pipelines.

📦 Real-World Use Case: Signing a Facturae XML

Your Spanish entity uses the **Spanish electronic invoice (ES)** feature to submit Facturae XMLs to the FACe portal.

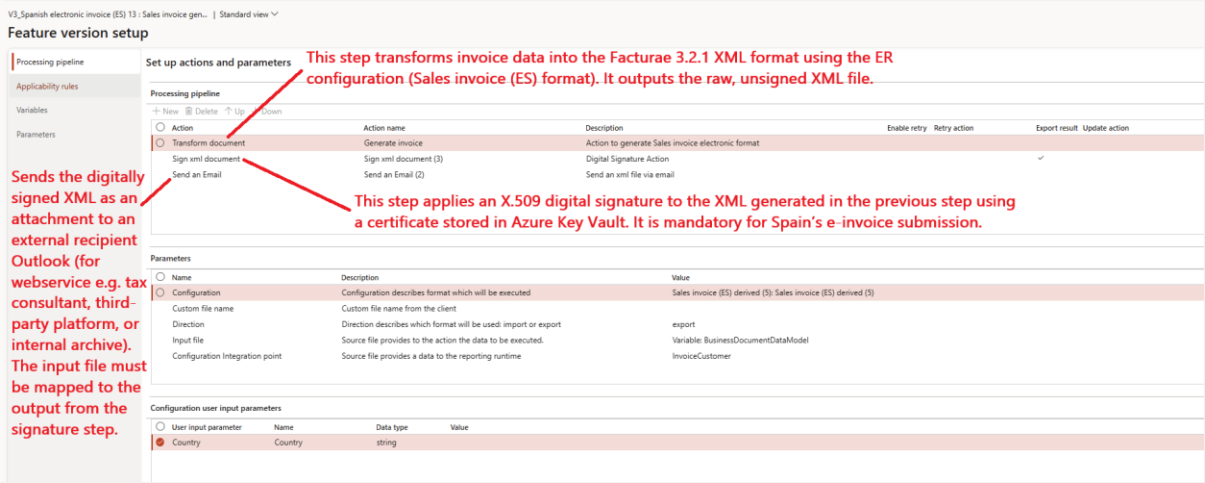
⚠️ Microsoft does not provide out-of-box web service submission for Spain. This pipeline exports and signs XML for external submission.

After **signing**, the file can be:

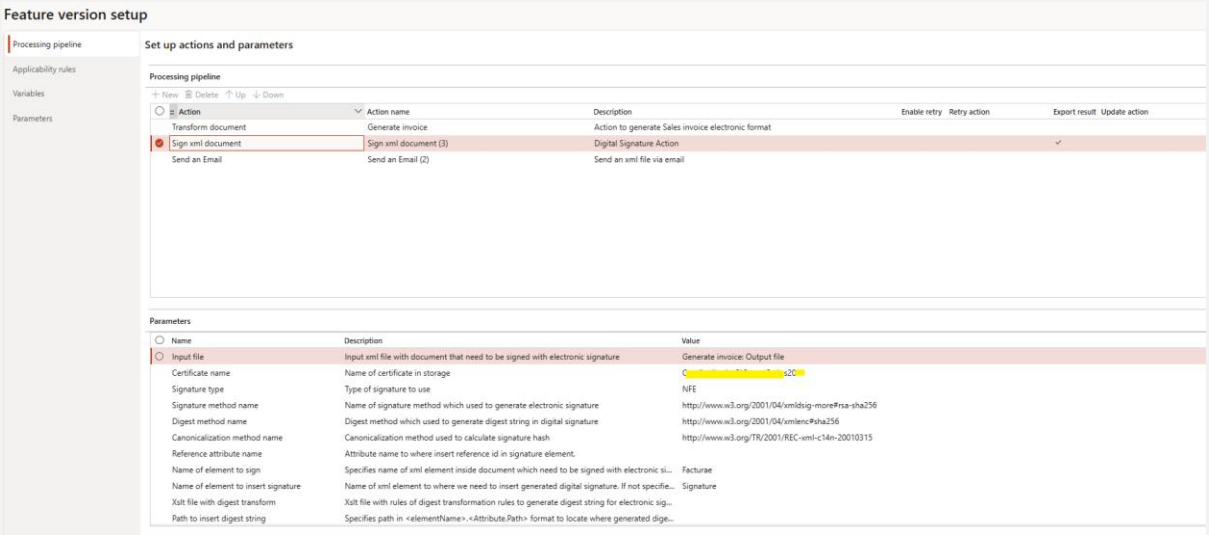
- Sent to **Azure Blob or SharePoint** for archiving or external pickup
- Posted to **Azure Logic App** to connect with a government-approved intermediary (ISV or FACe)
- Processed offline and manually submitted via Spain's **FACe portal**

Here's what the Globalization Studio pipeline looks like:

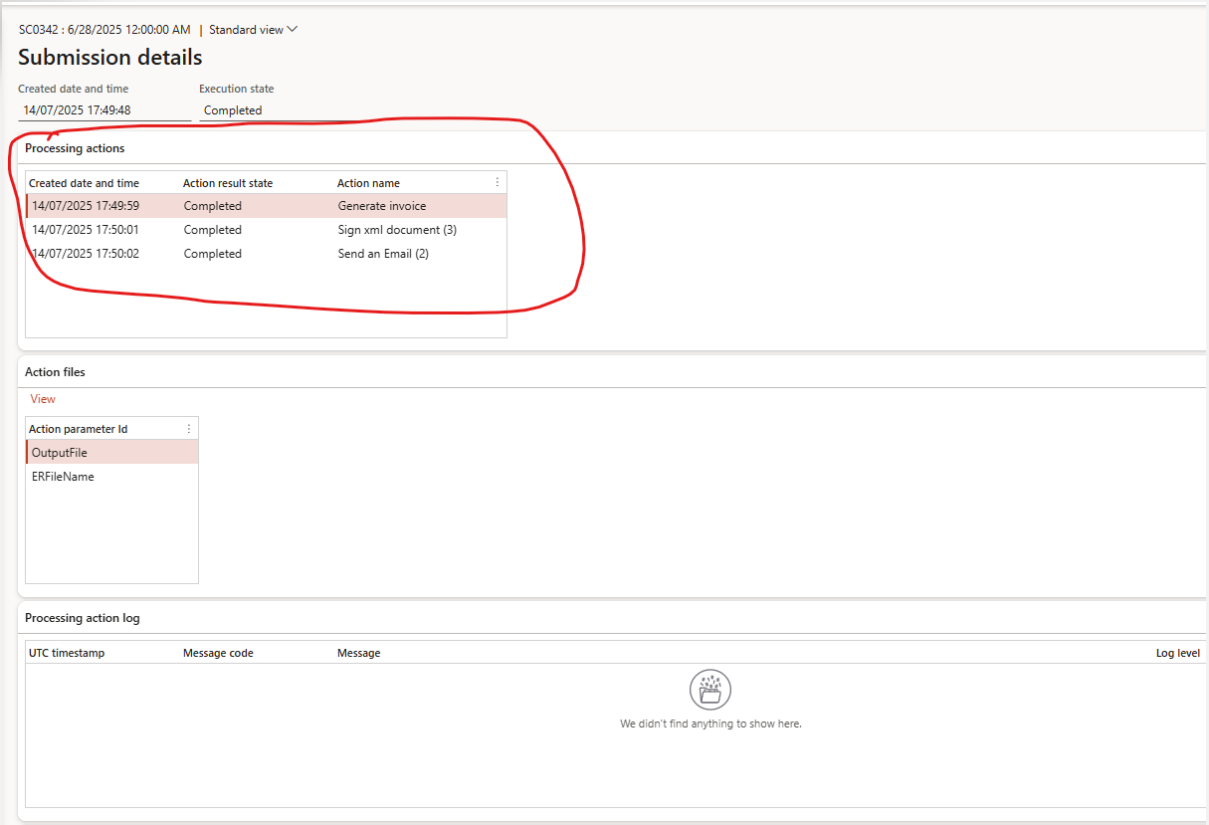
1. **Configure Feature Pipeline** → ER format creates the Facturae XML



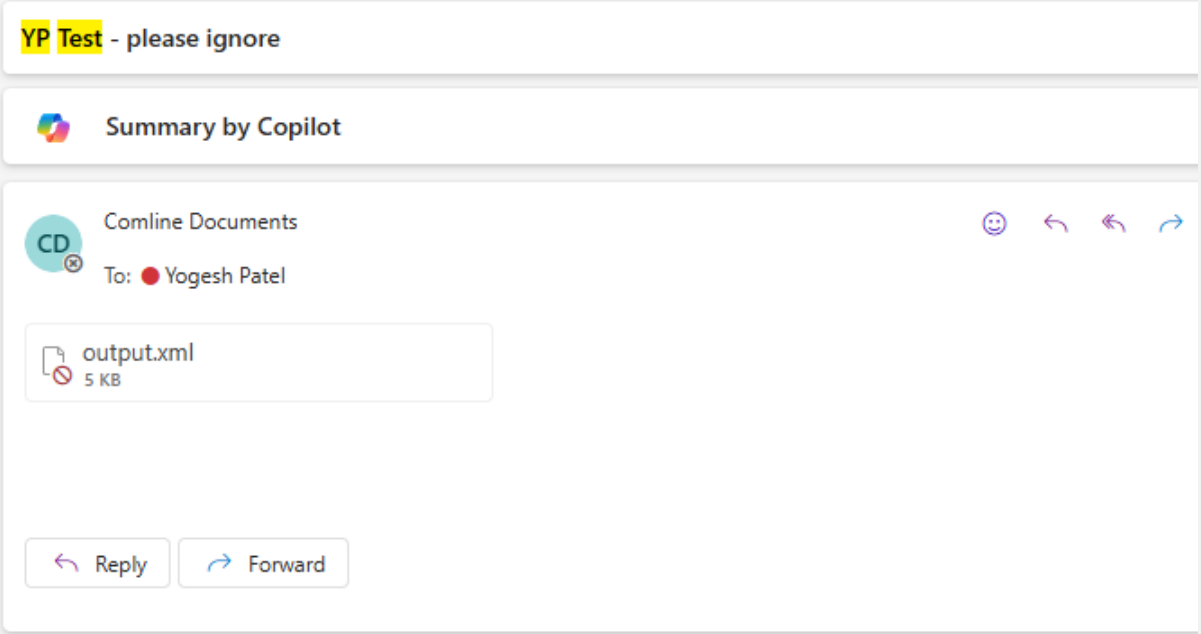
2. **Sign document Step** → XML is signed using a Key Vault certificate



3. **Process Invoice** → Submit Electronic Document and check processing log



4. **Exported result**



5. **Verify XML file digitally signed**

```
      <TotalPaymentsOnAccount>0.00</TotalPaymentsOnAccount>
      <TotalExecutableAmount>[redacted]</TotalExecutableAmount>
    </InvoiceTotals>
  ▼<Items>
    ▼<InvoiceLine>
       <ItemDescription>Brake Pad [redacted]716</ItemDescription>
       <Quantity>1.0</Quantity>
       <UnitOfMeasure>EA</UnitOfMeasure>
       <UnitPriceWithoutTax>[redacted]00</UnitPriceWithoutTax>
       <TotalCost>[redacted]0</TotalCost>
       <GrossAmount>[redacted]3</GrossAmount>
     ▼<TaxesOutputs>
       ▼<Tax>
          <TaxTypeCode/>
          <TaxRate>[redacted]0</TaxRate>
        ▼<TaxableBase>
           <TotalAmount>[redacted]3</TotalAmount>
         </TaxableBase>
        ▼<TaxAmount>
           <TotalAmount>[redacted]2</TotalAmount>
         </TaxAmount>
       </Tax>
     </TaxesOutputs>
    </InvoiceLine>
  </Items>
</Invoice>
</Invoices>
<Signature/>
▼<Signature xmlns="http://www.w3.org/2000/09/xmldsig#">
  ▼<SignedInfo>
     <CanonicalizationMethod Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
     <SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
   ▼<Reference URI="">
     ▼<Transforms>
        <Transform Algorithm="http://www.w3.org/2000/09/xmldsig#enveloped-signature"/>
        <Transform Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
      </Transforms>
      <DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
      <DigestValue>[redacted]zw1uqa/s=</DigestValue>
    </Reference>
  </SignedInfo>
  <SignatureValue>[redacted]/4M3BLVMcxn8bwajRUc[illegible]pw
  </SignatureValue>
 ▼<KeyInfo>
   ▼<X509Data>
      <X509Certificate>[redacted]yZXNlbnRhY2nDs24wHhcNMj[illegible]
      </X509Certificate>
    </X509Data>
  </KeyInfo>
</Signature>
</Facturae>
```


Digital Signature Included in XML File

Step-by-Step: How to Set Up Digital Signing

Step 1: Obtain a Valid Certificate

You'll need a .pfx certificate file issued by a trusted provider. It should:

- Be issued to your legal entity
- Support electronic signing
- Include a private key

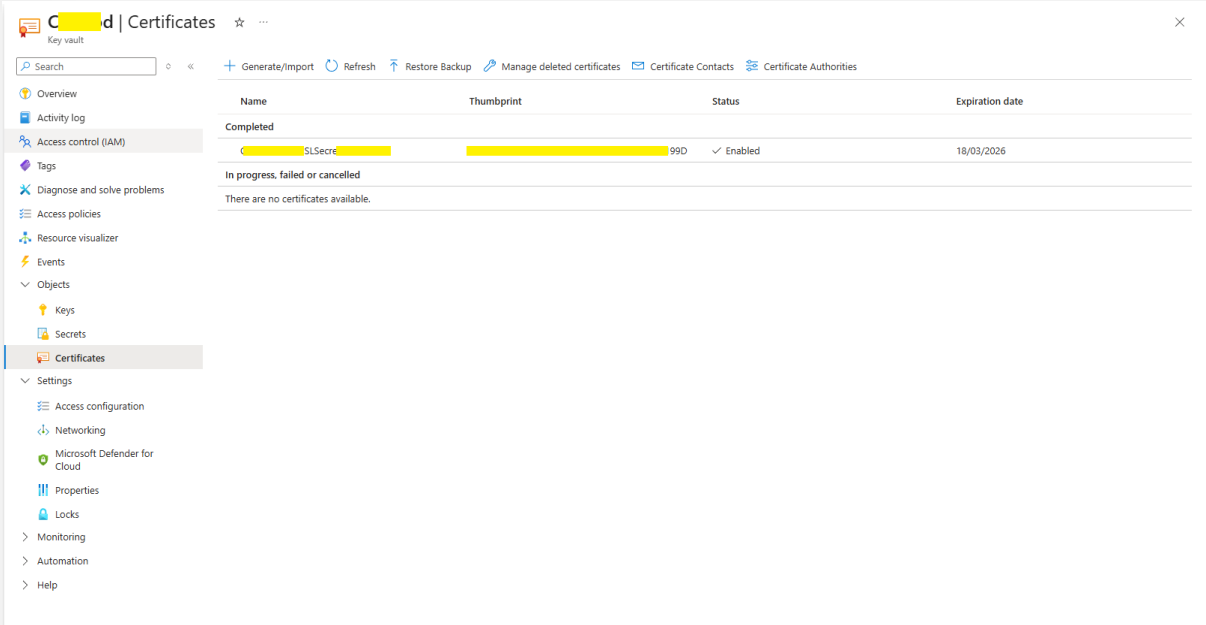
Name	Type	Size
Electronic Signature	Personal Information Exchange	

Step 2: Import the Certificate to Azure Key Vault

In Azure:

1. Open your **Key Vault**
2. Go to **Certificates > Import**
3. Upload the .pfx file
4. Name it clearly (e.g., FacturaeCert2025)

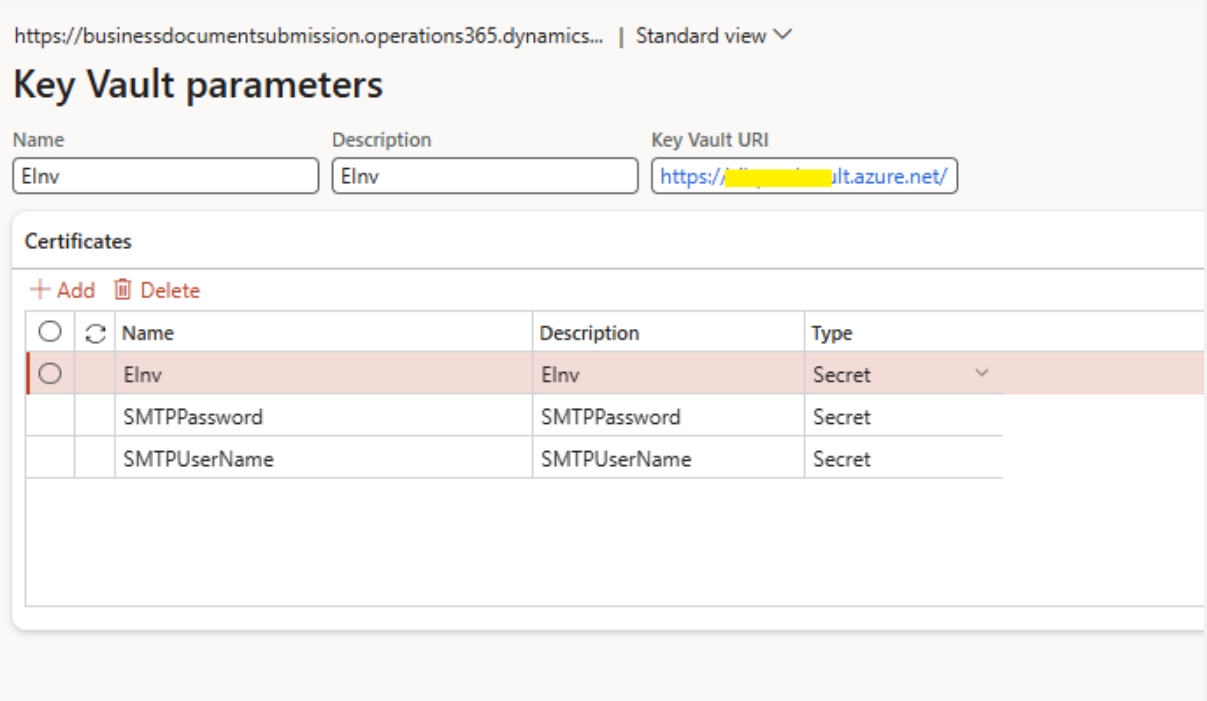
5. Add an **access policy** to allow Dynamics 365 access using your Azure AD app registration

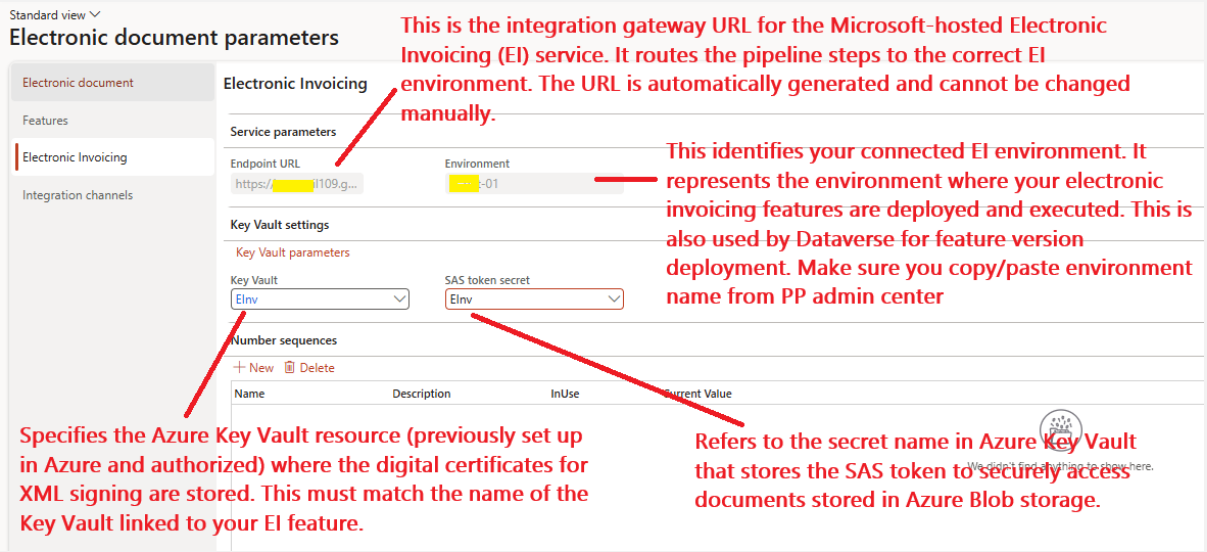


Step 3: Register Key Vault in Dynamics 365

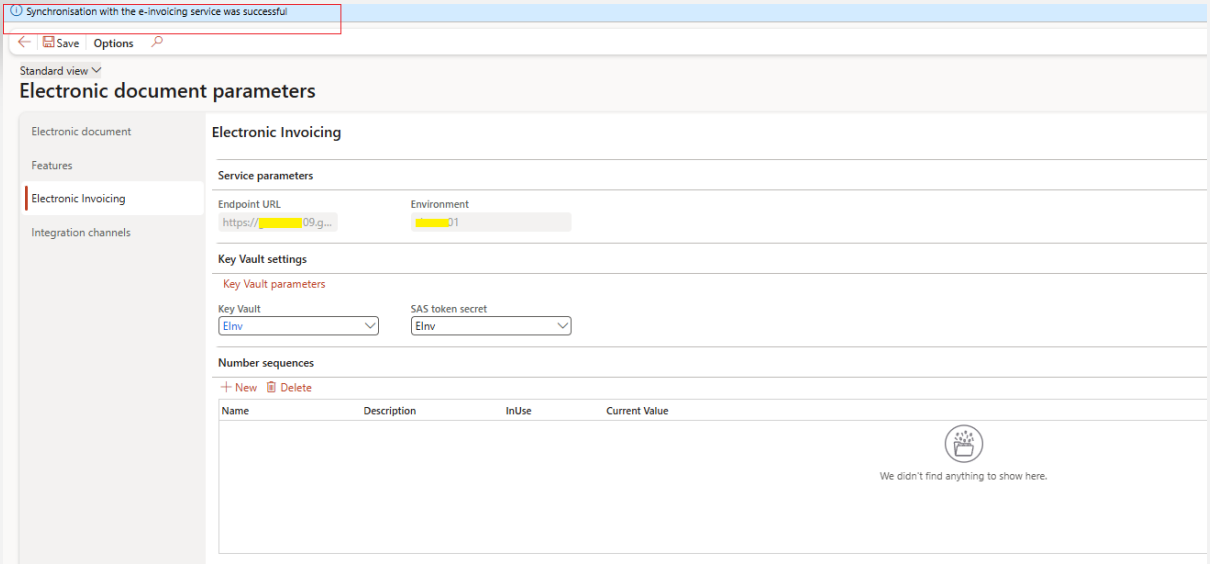
In D365:

1.
2. Go to **Globalization Studio > Electronic document parameters > Electronic Invoicing**
3. Open the **Key Vault Parameter** under Key Vault Settings, Add your details

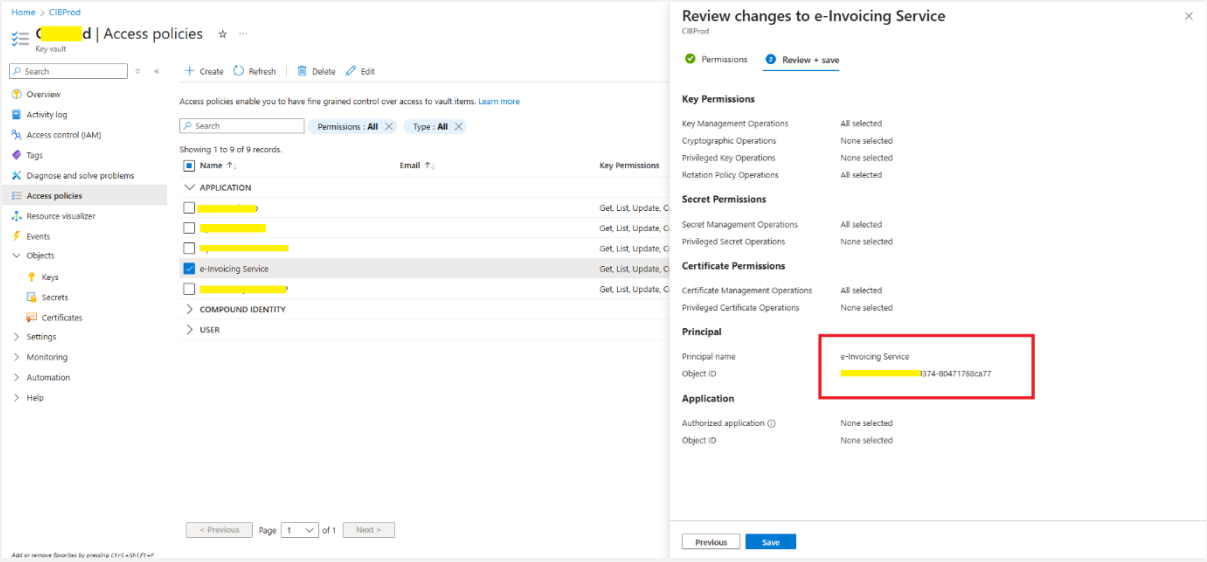




4. Test the connection to confirm access



Remember you provide e-invoice service access to key vault



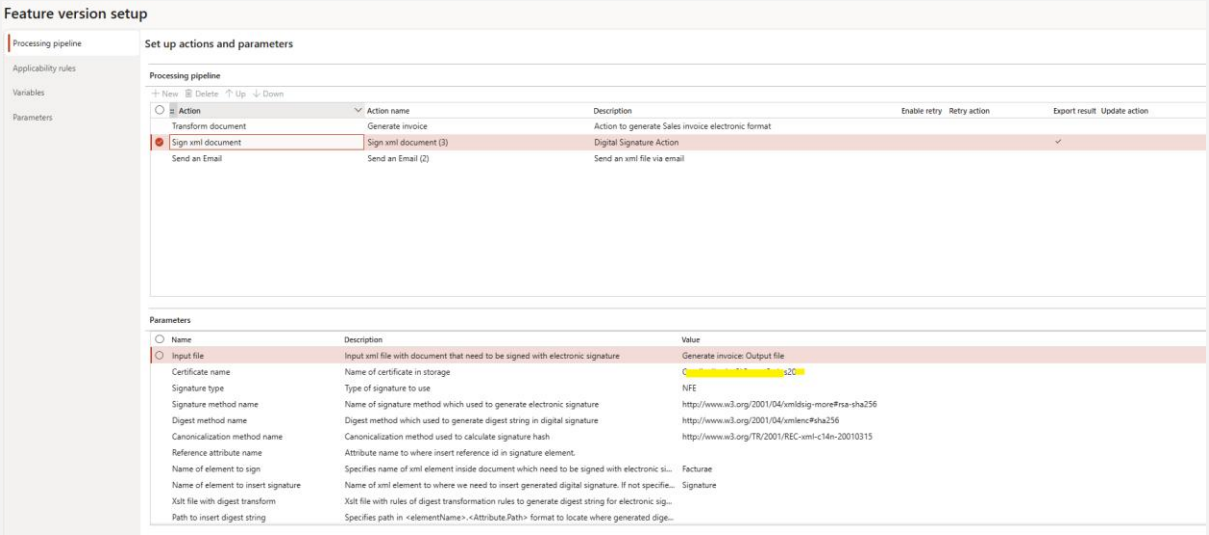
☑ Step 4: Add a "Sign Document" Step in the Pipeline

Inside the **electronic invoice feature** (e.g., Spanish electronic invoice):

1. Open the **Feature Setup > Processing Pipeline**
2. Add a new action of type **Sign document**
3. Configure:
    - **Input file**: Output from previous ER format step

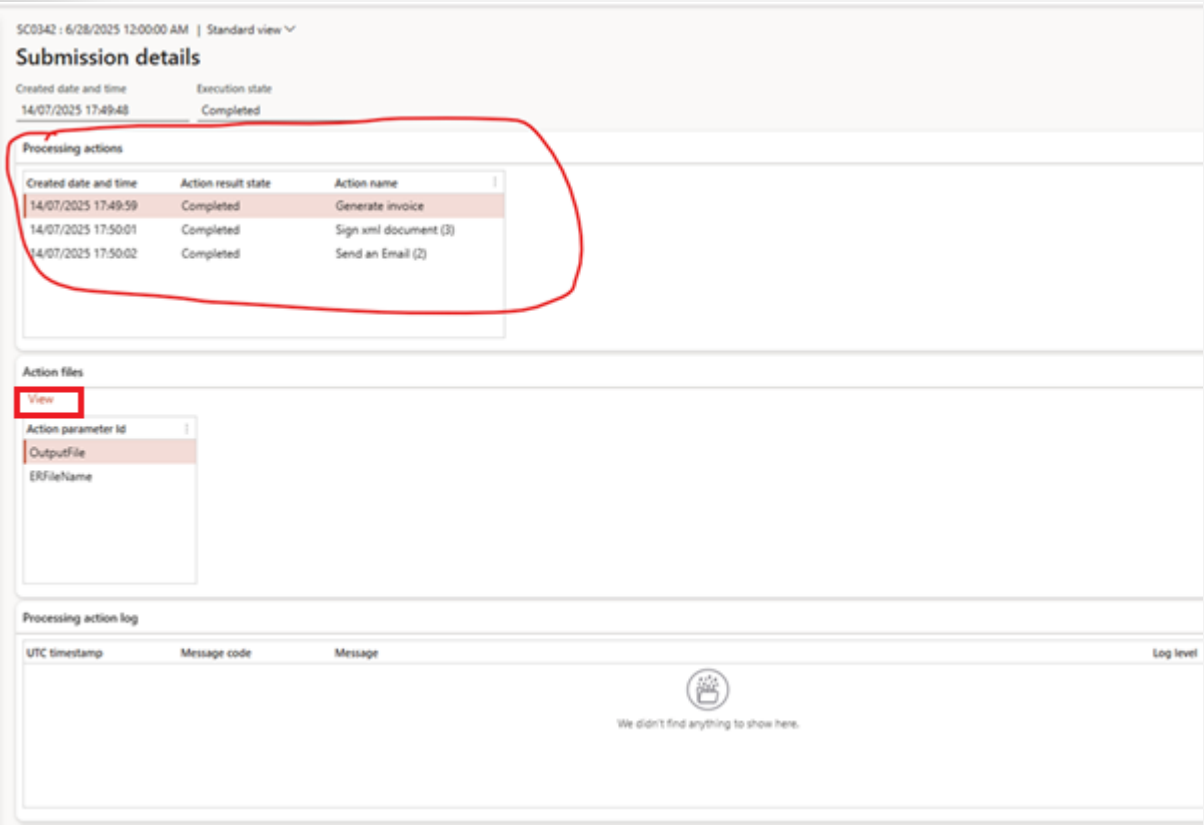
- **Certificate name**: From Key Vault (e.g., FacturaeCert2025)
- **Signature type**: XmlDsig
- **Digest method**: sha256
- **Canonicalization**: c14n

Every time the pipeline runs, the invoice XML is signed before it is exported



Where to View Signed Output

Location	What You'll See
Electronic messages > Attachments	The signed XML (with <ds:Signature> block)
Submission logs	Confirmation of signing success/failure
Outlook or Azure Blob or SharePoint (if configured)	Archived signed files by adding step in pipeline (**Azure File share, Save file to sharepoint)**



How This Differs from Spain's SII (MTD) Signing

Spain's SII does not sign the XML itself. Instead:

- The **electronic message transport layer** uses a certificate to sign the HTTP request
- The signing logic is configured in the **Send message action** within **Electronic message processing setup**
- No signature appears in the payload XML
- You do not use Electronic Document Parameters or Globalization Studio pipelines for SII

📘 For full SII setup, see GS518 – Electronic Messaging for SII

Tips for Managing Digital Certificates

Tip	Why It Helps
Name certificates clearly	Easier to reference in multiple environments
Use separate certs per country	Simplifies compliance and audit readiness

Tip	Why It Helps
Rotate certificates early	Avoid failed submissions due to expiry
Always test with a dummy cert in UAT	Avoids blocking production workflows

---

Summary

Digital signatures in Globalization Studio help ensure your invoice or document is:

- Authenticated
- Legally valid
- Protected from tampering

For **Electronic Invoicing features**, signing is managed through the pipeline and Azure Key Vault.

For **SII (MTD)** and similar integrations, signing is handled at the **message transport level** through Electronic Messaging.

Related Articles in This Series

- GS507 – Electronic Invoicing Overview
- GS510 – Reusing and Adapting Microsoft Features
- GS514 – Document Routing and Storage
- GS516 – Connecting to Government APIs

Coming Up Next

In **GS516 – Connecting to Government Portals**, you'll learn:

- How to configure submission steps to web service endpoints like FACe or SDI
- How retry logic, responses, and error handling work
- How to use Microsoft's prebuilt integration templates

📖 [Continue to GS516 → Government Web Service Submission →]