Bug Bounty Getting Started Instructions Visma AutoInvoice

Attack Surfaces in Scope

SOAP API

Currently the most used interface to our system.

Documentation

https://documentation.maventa.com/soap-api/
(all methods have sample scripts that can be used/benchmarked)

WSDL endpoint

https://testing.maventa.com/apis/denver/wsdl

API endpoint

https://testing.maventa.com/apis/denver/api

Common Methods

- register_with_password to create new companies
- configure_company to update company settings
- invoice_put_invoice_with_metadata to send invoices
- invoice_list_inbound_between_dates to list incoming invoices
- inbound_invoice_show to download an invoice

Example SoapUI Project

https://vismabugbountyprod.z16.web.core.windows.net/Maventa-Denver-soapui-project.xml

REST API

Newer API. Usage is growing fast.

Documentation

https://documentation.maventa.com/rest-api/
https://ax-stage.maventa.com/swagger/#/

Endpoint

https://ax-stage.maventa.com

Common Methods

https://ax-stage.maventa.com/swagger/#!/oauth2/postOauth2Token authentication

https://ax-stage.maventa.com/swagger/#!/documents/postV1Documents for sending documents
https://ax-stage.maventa.com/swagger/#!/documents/getV1Documents for downloading documents
https://ax-stage.maventa.com/swagger/#!/company/postV1CompanyNotifications register webhook for callback notifications on invoice events

Main Web UI

Web UI, parts of it already replaced with the newer Embeddable UI described below.

Login using the credentials described in the “User Accounts and Sample Data” chapter.

Endpoint

https://ai-testing.maventa.com/ (also https://testing.maventa.com, same application with different UI branding).

Embeddable UI

Embeddable UI, using the REST API for all data operations.

Authenticate by providing parameters in the URL in one of these formats:

- https://autointerface.stag.visma.net/invoices?user[user_api_key]=ENTERAPIKEYHERE&user[company_uuid]=ENTERCOMPANYUUIDHERE&profile=autoinvoice
- https://autointerface.stag.visma.net/invoices?token=TOKEN_U_GET_FROM_AX_API_OAUTH2_METHOD

The first authentication method with the user params in the URL is being deprecated but currently still works.

Endpoints (both point to the same application)

https://autointerface.stag.visma.net
https://autointerface-embeddable-stage.maventa.com

User Accounts and Sample Data

Registration

Create your own test account on https://ai-testing.maventa.com/registrations

Organization number of the company can on stage be faked/made up. Some countries might have domestic syntax validation (e.g. modulus checks at the end) enabled.

You need to use an email address you have access to in order to be able to activate the user and get access to the passcode needed on login.
New companies are created in an unverified state which means they require a strong authentication of the user before allowing access. **This requirement is currently only**

enabled for Finnish companies (country code FI) on the testing site. Circumventing this protection will be rewarded.

To get access to the company UUID and user api keys needed for API access, navigate to https://ai-testing.maventa.com/settings after login.

API usage and registration

Several of the API methods on both SOAP API and REST API, including registration, require a vendor API key (identifies the software using the integration). Below are some vendor keys for testing purposes:

BugBounty Vendor 1 key (**trusted**): `bd83b469-0b62-4451-9fa7-56f2394fd190`
BugBounty Vendor 2 key (**untrusted**): `4fc526aa-1f6c-4cb8-be91-14cb9b0dea26`

Registration with a **trusted** key allows using the account immediately (login to UI, send invoices through the API etc.). Registration with an **untrusted** key will require strong identification of the user before accessing UI or sending invoices etc. The strong identification is done using Visma Sign which sends an email invite to sign an agreement. On the testing site, the following information can be used to do the actual signature:

Choose identity provider: “Test identification”, logo looks like this:

Username: testtest@test.test
Password: xO70MVYD0CXaKcQ2

Sample Invoice Files

https://github.com/OpenPEPPOL/documentation/blob/master/PostAward/InvoiceOnly4A/Archive/20140301-PEPPOL_BIS_4A-400-AppendixA-Use%20Case%20test%20files.zip