Valid Value Technical background/IT/data information

Our product/software consists of 3 components:

- The **Valid Value model** (an excel file with VBA/macros, a .xlsm file)
- Our standard reports (Word, PowerPoint)
- The **Valid Value add-in** (an installation file, built in C#)

The Valid Value model

Our model is an **Excel/.xslm file**, which can be placed and opened by the user anywhere – on a hard drive, a network drive, sharepoint etcetera. Because it is an ordinary excel file, the user can also make endless copies. We have secured the file to prevent users from breaking the file on the one hand, and to protect our intellectual property on the other. The file is therefore "just excel", but can also be seen as software, since the Excel file is quite closed and we have provided it with its own menu (named *Valid Value Model*) and underlying VBA code (macros). When using the file, our add-in is required to be able to open and use the file.

For example, we use the macros in the model to:

- To be able to close and open sheets/rows/columns in the file
- Perform certain calculation processes (scenarios)
- Correctly add/remove rules in the model
- Interact with the add-in to:
    - Verify that a user has a valid license
    - Retrieve support data from api.validvalue.com

Of course, we are looking at possible alternatives that are possible within Microsoft Office, but the things we are doing now with VBA/macros can simply only be done with macros at the moment (given the conditions). The legitimate key question is, of course, how we guarantee, and how you (as well as all our other customers) can trust that our VBA code is secure.

We sign our VBA code with a certificate with every update of our model. This certificate is issued by Sectigo (https://www.sectigo.nl/), one of the largest certificate suppliers in the world.
Our models are secured in such a way that no one can access the VBA code. In the unlikely event that someone succeeds in doing so and changes one letter in the code, this is immediately detected and the certificate is invalid. Within Excel, it is possible to set up in the security centre that, for example, macros are only accepted in models that include a signed and valid certificate. This happens at several of our customers. We are responsible for our codes, which are created and verified entirely in-house by ourselves. These VBA codes are necessary for Valid Value to work.

Macros & ASR / Microsoft Defender

In order for our model to work in conjunction with Microsoft Defender / ASR, it may be necessary to modify these rules:

- Make sure the models are in a "non-protected directory" (so macros can be executed)
- Excel settings are custom (enable macros, auto restore off and disable auto-save)
- Within the Microsoft baselines, turn off the following:
    - Block Win32 API calls from Office macro
    - Block execution of potentially obfuscated scripts (js/vbs/ps)
    - Block JavaScript or VBScript from launching downloaded executable content

The standard reports
These are normal Word/PowerPoint files in which we have some intelligence. As a result, these files, in combination with our add-in, can offer extra functionality – in particular, the smart linking of a table in Excel to an image in Word, for example. This is not VBA/macros, all functionality is in the add-in.

De Valid Value Add-in
We develop this add-in ourselves in C#. The add-in adds an additional toolbar called Valid Value Tools to Excel, PowerPoint, and Word. That add-in offers a number of additional features in Excel and makes it possible to create dynamic links between Excel and PowerPoint files (and Excel and Word files). Of course, our installation file is also digitally signed by a Sectigo certificate with which we sign our VBA codes.

Installation issues?
When installing our add-in, it can happen (*especially in server/RDP setups*) that the add-in does not load/come to a user after installation. This is recognizable by the fact that in Excel/Word/PowerPoint there is no toolbar "Valid Value Tools" – the program is installed, but a number of registry keys are missing for the user.
The most convenient way to solve this is to install the add-in per user via a Group Policy: https://www.add-in-express.com/creating-addins-blog/automatic-installation-windows-group-policy/

A backup/alternative is to manually add the add-in to Excel for the user once. This will create the missing registry keys. This can be done by:

- Install the add-in (you can check the installation by going to "Install or uninstall programs" within Windows and see that the add-in is there)
- Go to Excel and choose File->Options->Add-ins – the add-in will not be visible here
- At the bottom, choose Manage: "COM Applications" -> Start
- Choose C:\Program Files (x86)\Valid Value BV\Valid Value Add In\ and choose adxloader64.dll (default: 64-bit systems) or adxloader.dll (32-bit).
- After this, the **Valid Value Tools** bar should load in Excel, Word, and PowerPoint. Please note that the user still needs to log in to this toolbar to activate and use the add-in.

Data confidentiality
Our excel model "communicates" with our server, https://api.validvalue.com/, through our add-in. That URL is the only external endpoint that our software connects to. The only data we communicate with our server:

- License details – login/logout in our add-in
- Support data – we retrieve this data from our server, depending on user input
- Basic User Statistics

Under no circumstances will we communicate with another endpoint from our software. We also do not send any data that has been entered into the model to the outside world – all data remains in your local excel file. The only exception is the parameters we need to provide the support data – country, sector and valuation date.