

# The Elevation tab

## Run as:

The Elevation tab is an important part of what makes MarvelClient Upgrade so different and much more successful than any other solution.

In most environments, end users do not have administrative rights on their machines (or must confirm a Windows User Account Control (UAC) dialog to grant an installer corresponding rights).

The Run as section on the Elevation tab takes care of various types of elevation:

- [Run via MarvelClient MCUUpgrade as Admin](#)  
In this case, MarvelClient deploys at least the Upgrade Configuration, personalizes it for the respective user and client (think: data directory and client language, for example) and performs elevation via a multi-level encrypted key file.
- [Run via MarvelClient MCUUpgrade Service](#)  
In this case, MarvelClient deploys at least the Upgrade Configuration, personalizes it for the respective user and client (think: data directory and client language, for example) and performs elevation via the panagenda Installer Service.  
For further details on the panagenda Installer Service, see below.
- [Deploy and configure via MarvelClient and Run via your own solution](#)  
In this case, MarvelClient deploys at least the Upgrade Configuration, personalizes it for the respective user and client (think: data directory and client language, for example), but leaves elevation and execution to your own solution.
- [Static configuration and run via your own solution](#)  
In this case, MarvelClient creates a static Upgrade Configuration (IMPORTANT: ALL your clients must match the specified program, data etc. directories!) for elevation and execution with your own solution.



### IMPORTANT

Your upgrade will be **much** more successful, if you let MarvelClient for Notes deploy and configure your MCUUpgrade.ini - that is because almost ALL our customers have clients that do not correspond to corporate standard.  
Any client with different directories will either fail to upgrade or experience loss of data or functionality for the end user.  
Together with MarvelClient Essentials/Basic/Upgrade, you do not have to worry about "what is out there" but can simply rely on MarvelClient to take care of it.

## Install panagenda Installer Service if missing:

Installs the panagenda Installer Service for easy use in your subsequent upgrade project.

The panagenda Installer Service is a Windows Service, which *exclusively* accepts elevation requests from binaries with an official **Windows Extended Validation (EV) Certificate** from panagenda.

## Enable Whitelisting (mandatory when using the panagenda Installer Service)

When running an Upgrade via the panagenda Installer Service, all instructions in MCUUpgrade.ini are signed to prevent tampering with any instruction.

Whitelisting is also enforced (=mandatory), when you deploy your own RSA 4096 bits MCUUpgrade\_PublicKey.sig file to %ProgramFiles(x86)%\panagenda\MCUpgrade (make sure that end users have read permission only!).

For further details on Whitelisting, see [The \(MC\)KeyfileEditor for Whitelisting](#).

## Admin rights (only visible with Elevation via MCUUpgrade as Admin):

- [NOT all end users have admin rights](#) (default)  
In most environments this is the right setting to choose and will require you to provide an encrypted key file with account names and passwords.
- [ALL end users have admin rights](#)  
In case all your end users have admin rights AND it is ok for you, if your end users have to confirm the UAC elevation dialog, you can choose this option for easy testing and rollouts without any elevation credentials.  
IMPORTANT: ALL your end users must have admin rights then!

## Disable Local Admin:

If you do not provide a key file, you must leave this option unchecked.

Disable this option, if you do not want to provide a key file (or just want to test with UAC elevation).

#### Disable Local Admin Fallback:

In case elevation fails with the provided key file, you can enable or disable fallback to attempting the Upgrade as end user for those users who have admin rights (this may raise a UAC prompt then).

#### Elevation files(s):

Depending on your above configuration, this field holds no files, or the MCUpgrade\_PublicKey.sig and/or the MCUpgrade.mcu (key) file.

For further details on how to create either file, see [The \(MC\)KeyfileEditor for Elevation](#).