

Use root certificates included with FileMaker

If you like to have SSL with security, you need to verify the certificate. FileMaker comes with a root certificate file, so we can refer CURL to that file and enjoy the checks against the given list of root certificates:



```
Set Variable [ $curl ; Value: MBS("CURL.New") ]
#
If [ MBS("IsIOS") ]
    # look for path to certificate file
    Set Variable [ $path ; Value:
Substitute(MBS("Folders.UserDocuments"); "/"
Documents"; "/Library/Application Support/FileMaker/iOS/1.0/root.pem") ]
    If [ MBS("Files.FileExists"; $path) ]
        # use in CURL
        Set Variable [ $r ; Value: MBS( "CURL.SetOptionCAINFO"; $curl;
$path) ]
    Else
        Show Custom Dialog [ "File not found." ; $path ]
    End If
Else If [ MBS("IsMacOSX") ]
    # look for path to certificate file
    Set Variable [ $path ; Value: Substitute(MBS("Folders.UserDocuments"); "/"
Documents"; "/Library/Application Support/FileMaker/Shared/17.0/root.pem") ]
    If [ MBS("Files.FileExists"; $path) ]
        # use in CURL
        Set Variable [ $r ; Value: MBS( "CURL.SetOptionCAINFO"; $curl;
$path) ]
    Else
        Show Custom Dialog [ "File not found." ; $path ]
    End If
Else If [ MBS("IsWindows") ]
    # look for path to certificate file
    Set Variable [ $path ; Value: Substitute(MBS("Folders.UserDocuments");
"\Documents"; "\AppData\Local\FileMaker\FileMaker Pro
Advanced\17.0\root.pem") ]
    If [ MBS("Files.FileExists"; $path) ]
        # use in CURL
        Set Variable [ $r ; Value: MBS( "CURL.SetOptionCAINFO"; $curl;
$path) ]
    Else
        Show Custom Dialog [ "File not found." ; $path ]
```

```

    End If
Else If [ MBS("IsLinux") ]
    # look for path to certificate file
    Set Variable [ $path ; Value: "/FileMakerData/Extensions/OpenSSL/
root.pem" ]
    # use in CURL
    Set Variable [ $r ; Value: MBS( "CURL.SetOptionCAINFO"; $curl; $path) ]
End If
#
# Start new session
Set Variable [ $URL ; Value: "https://www.filemaker.com" ]
# Set URL to load (HTTP, HTTPS, FTP, FTPS, SFTP, etc.)
Set Variable [ $result ; Value: MBS("CURL.SetOptionURL"; $curl; $URL) ]
// Set Variable [ $result ; Value: MBS( "CURL.SetOptionCertInfo"; $curl; 1 ) ]
# Do transfer now:
Set Field [ Placeholder::result ; MBS("CURL.Perform"; $curl) ]
# Check result
Set Field [ Placeholder::Container ; MBS("CURL.GetResultAsText"; $curl; "UTF8")
]
Set Field [ Placeholder::result ; MBS("CURL.GetDebugAsText"; $curl) ]
// Set Field [ Placeholder::result ; MBS("CURL.GetCertinfo"; $curl; "UTF8") ]
# Cleanup

Set Variable [ $result ; Value: MBS("CURL.Cleanup"; $curl) ]

```

This script works fine with MacOS, Windows, Linux and iOS. If needed you can request certificate information via [CURL.SetOptionCertInfo](#) and see all the details from certificates in the text given by [CURL.GetCertinfo](#) function.