

FILEMAKER® SERVER 17 AND SSL CERTIFICATES: CONFIGURATION AND USE



By:

Wim Decorte

—and—

Steven H. Blackwell



Version 1.0v1

FileMaker Business Alliance Platinum Members are independent entities
without authority to bind FileMaker, Inc.
and FileMaker, Inc. is not responsible or liable for their actions.

The views and recommendations expressed in this White Paper are solely those
of the authors and may not necessarily reflect those of FileMaker, Inc.

FileMaker, FileMaker Go and the file folder logo are registered trademarks of FileMaker,
Inc. in the U.S. and other countries. FileMaker WebDirect and FileMaker Cloud are
trademarks of FileMaker, Inc.

© Copyright Wim Decorte and Steven H. Blackwell, 2018.
All rights reserved under both International and Pan-American Conventions.
Permission granted to users of FileMaker products
to distribute within their own organizations.

TABLE OF CONTENTS

Introduction.....	1
What Is SSL?	2
What Does SSL Protect? Why Do We Need To Utilize It?	3
What Does SSL NOT Protect?	4
How To Obtain A SSL Certificate.....	5
How To Generate A Certificate Signing Request.....	5
Buying The SSL Certificate.....	6
Importing A Custom SSL Certificate Into FileMaker Server.....	8
A Side Note: HSTS Connectivity	12
Testing For Secure Connections To FileMaker Server.....	12
The Importance Of DNS.....	14
Multi-Machine Deployments	17
Other CLI Options	18
Summary	19
Acknowledgements.....	20
About The Authors	21

FILEMAKER® SERVER 17 AND SSL CERTIFICATES: CONFIGURATION AND USE

By:

Wim Decorte and Steven H. Blackwell

Version 1.0v1

This White Paper is one of a three-part series released with the new FileMaker® Server 17. The other two papers are *FileMaker® Server 17: Admin Console, Admin CLI, and Admin API* and *FileMaker® Server 17: Server Monitoring Functionality*.

They can be found at this link:

<http://community.filemaker.com/docs/DOC-8939>

The principal purposes of all three White papers are to inform Developers, Server Administrators, and end-users of some significant features of FileMaker Server, both long-time ones and new ones, and to identify some new behaviors and new locations for managing various aspects of FileMaker Server and related elements.

FileMaker Server runs on various Windows Server and macOS Operating Systems. It also runs in FileMaker® Cloud, although that deployment option is not a focus of these papers.¹ The new behaviors and processes for managing FileMaker Server 17 are very different than those in prior versions, including Version 16. FileMaker, Inc. replaced the long-time FileMaker Server Admin Console with a totally new Console. Additionally, they have expanded to the Command Line Interface (CLI) and introduced² the Server Admin API based on the REST capabilities first introduced in FileMaker® Server 16 with the Data API.

¹ While not the focus of this paper, all of the fundamentals and much of the SSL functionality is the same on FileMaker® Cloud. When in doubt, contact FileMaker Support or ask a question on community.filemaker.com.

² The Admin API was first launched with the FileMaker Cloud version released in October 2017.

All three of these papers will follow the same general structure and format:

1. Brief description of the topic(s) to be covered in the specific paper
2. Description of concept underlying the feature
3. Description of why it is important
4. Description of how to configure
5. Description of how to use
6. Summary at the end

We strongly recommend that Developers and Server Administrators carefully review all the documentation that accompanies FileMaker Server 17 as well as these three papers. Many Best Practices from prior versions remain in effect. However, most of them require new methods to implement.

WHAT IS SSL?

A critical element for effective business use of data of every type is that all parties must have assurance that the Confidentiality and Integrity of data are maintained. Secure Socket Layers (SSL) and the related Transport Layer Security (TLS) are cryptographic protocols that provide communication security over a computer network, whether that be a Local Area Network (LAN), a dedicated Wide Area Network (WAN), or the public Internet. While we often refer to this protocol as SSL, and we do so in this White Paper, in reality that has been superseded by the more modern TLS, specifically by TLS 1.2. For simplicity sake we will continue to use the term “SSL” as it still is the most common term for this functionality.

SSL encryption works with what is called a *key pair*: a *private* key and a *public* key. Together these keys form the basis for encrypting the data in transit across the network. A SSL Certificate also contains an element generally called the *subject* that establishes and validates the identity of the server transmitting the data.

FileMaker Server uses SSL encryption for two purposes: **encryption of data in transit** and **verification of server identity**. The latter function helps to prevent what are called man-in-the-middle attacks. As that name suggests, this is a situation where an attacker could impersonate the identity of a server and steal or alter data. Encryption of data in transit is distinct from Encryption At Rest (EAR). EAR protects the physical binary FileMaker Pro Advanced file itself against various types of tampering. EAR is also a feature of the FileMaker platform, but we do not discuss it here. Similarly, the Field-Level Encryption functionality introduced with FileMaker® Pro 16 is another matter and not discussed here.

WHAT DOES SSL PROTECT? WHY DO WE NEED TO UTILIZE IT?

As noted above, SSL protects data in transit across the network by encrypting it. Software known as Packet Sniffers can capture and display all network data. If those data are not encrypted, then whoever uses a packet sniffer can read the unencrypted data, revealing confidential, proprietary information. Many FileMaker developers hold to the view that as long as all data are enclosed within the Local Area Network (LAN) and not traveling outside of it, that such encryption is unnecessary. We strongly believe otherwise; in fact, an unencrypted LAN can be a dangerous place. Consider some of these items:

- With the advent of Bring Your Own Device (BYOD), Attackers possibly can exploit any connected rogue, unsanctioned devices to gain network access. Through a process called *war-driving* (sometimes *war-chalking*), Attackers with sophisticated, high-powered antenna cruise through office parks or other areas looking for network access.
- Use of insecure Wi-Fi networks in malls, airports, coffee shops, and similar areas can likewise be a source of data leakage, even possibly when Virtual Private Networks (VPN's) are in use.
- Internal actors, malevolent or innocent, with authorized network access but who may not have authorized access to some data in a FileMaker Pro Advanced database, can intercept and read unencrypted traffic emanating from FileMaker Server. Internal actors are a significant source of data breaches.
- The concept of what constitutes an internal actor—or an insider—changes dramatically year after year. Organizations in your supply chain, for example, may be as much of an insider as a business owner or an employee when it comes to network or database access. Several prominent, significant, widely-reported data breaches likely started somewhere down the victim's supply chain.

In addition to protecting the data in transit, SSL also protects and verifies the *identity* of the FileMaker Server. That is, it *confirms that the server is who the server asserts that it is* much in the way that the password in the credentials dialog helps validate the identity assertion provided by the FileMaker Pro Advanced Account Name. This helps thwart the man-in-the-middle attacks. In such an exploit, the attacker (sometimes called a Threat Agent) intercepts, relays and *sometimes alters* communication between two parties who believe they are in *direct communication* with one another. The man-in-the-middle can then inspect and read that intercepted data. The certificate issued to the server helps to validate its identity and to thwart man-in-the-middle attacks. That is one of the reasons for using a fully verified certificate issued by an accepted Certificate Authority.

WHAT DOES SSL NOT PROTECT?

Enabling FileMaker Server to use SSL with a custom certificate encrypts data in transit between FileMaker Server and the native FileMaker clients such as FileMaker Pro Advanced, FileMaker WebDirect™, and FileMaker® Go. In addition, it encrypts data flowing between FileMaker Server Database Engine and the FileMaker Server Web Publishing Engine.

Both the Data API and the Admin API also function strictly through SSL protected HTTPS channels, and the language or technology you are calling from must support TLS 1.2. Connections to the Admin Console will always redirect to use HTTPS and SSL.

Using SSL does *not* encrypt data flowing between the user's browser and the Web Server used for custom web publishing through the PHP and XML APIs. That part of the traffic can be protected by configuring the web server separately to use the SSL certificate.

The FileMaker Server SSL certificate does also does not currently encrypt data flowing to and from a user or system connecting to FileMaker Server through ODBC or JDBC.

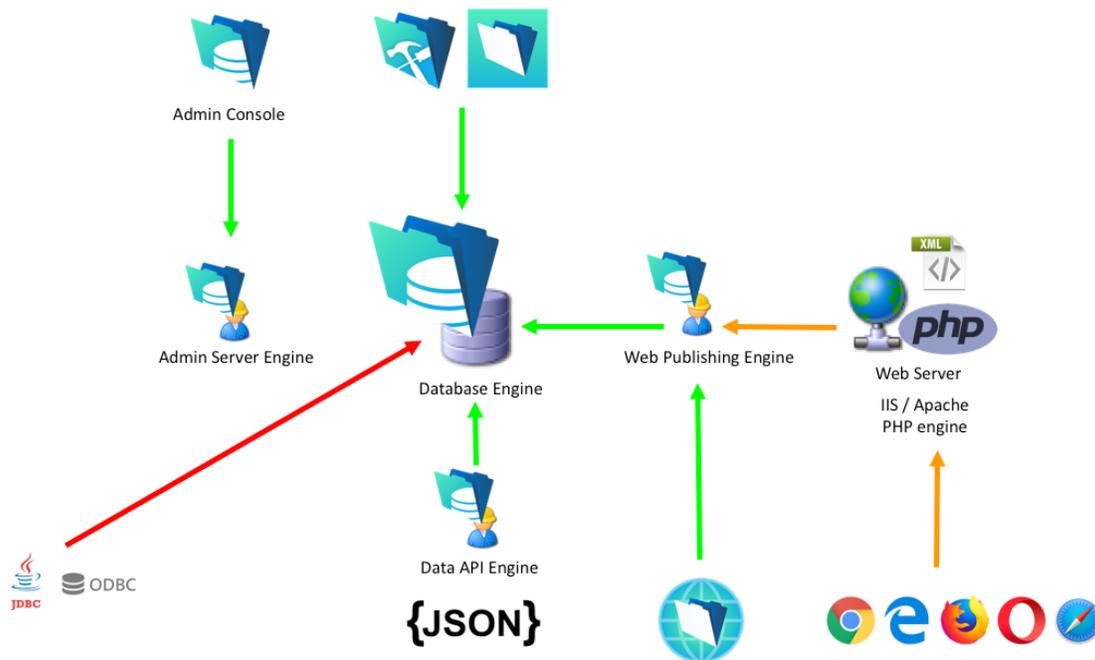


Figure 1. Encrypted Traffic FileMaker Platform Components.

HOW TO OBTAIN A SSL CERTIFICATE

The process for obtaining a SSL Certificate is straightforward, albeit somewhat cumbersome. FileMaker Tech Info 14176 describes the process. That article is located at the following link:

<https://thefmkb.com/14176>

or

<https://support.filemaker.com/s/answerview?anum=000025609> [new system]).

Basically, here are the steps:

- Acquire a domain (*e.g.* company.com). The certificate will be issued for that a Fully Qualified Domain Name (FQDN) entry for your FileMaker Server. A FQDN typically comes in the form of server.company.com. Or *.company.com for a wildcard certificate is used to cover multiple servers.
- Generate a Certificate Signing Request (CSR) and private key for that FQDN. **Safeguard the private key stringently.** It is paramount that the key not be compromised.
- Purchase a SSL Certificate from one of the approved Certificate Authorities (CA's) and have the certificate issued using the CSR you created in step 2.
- Import the Certificate into FileMaker Server.

The steps below assume that you have the proper domain ownership to have a certificate issued for it.

HOW TO GENERATE A CERTIFICATE SIGNING REQUEST

In order to obtain a Custom SSL Certificate, developers or administrators must send a document known as a *Certificate Signing Request* (CSR) to the Certificate Authority.

To create the CSR you must use the Command Line Interface command ***fmsadmin certificate create***. This command requires the unique Fully Qualified Domain Name for the server as well as a secret key that you generate.

For example:

```
fmsadmin certificate create server.company.com --keyfilepass secretkey
```

Note that the example here is the simplest form. Some SSL vendors will require more information in the CSR, so make sure to check what fields are mandatory for your preferred vendor and then include those fields in the command line.

A more complex example could look like this and includes information on the Organization (O), country, state and location (C, ST and L):

```
fmsadmin certificate create "/CN=server.company.com/O=Soliant Consulting  
Inc./C=US/ST=IL/L=Chicago"
```

Check the FileMaker Community space for tools provided by developers for generating these CSRs. (<http://community.filemaker.com/docs/DOC-8939>)

The command line process creates and places into the /FileMaker Server/CStore directory two files: **serverRequest.pem** and **serverKey.pem**. The **serverKey.pem** file and the secretKey string are needed for importing the file for the Certificate that the Certificate Authority returns to you. The contents of the **serverRequest.pem** file must be copied to the CA's certificate request site. *Use a text editor to do this, not a word processor.* Word processors will introduce spurious control characters into the file, causing it to generate errors.

In FileMaker® Server 16 and FileMaker® Server 15, creation of these files could be done from within the Console (<https://thefmkb.com/14176>). *In FileMaker Server 17, they must be done from the Command Line Interface.*

BUYING THE SSL CERTIFICATE

Now that you have a *Certificate Signing Request* (CSR) you can use it to have the SSL certificate issued by a Certificate Authority (CA).

What is a Certificate Authority? A CA is a trusted third-party organization that issues or signs certificates attesting to the ownership of a public cryptographic key by the owner or subject of the certificate. This allows validations of identity assertions and authenticity of public-key encryption for those accessing the sever where the certificate resides.

The certificate generation process rests upon a trust basis with the CA.

- °You trust the root-level certificate belonging to the CA
- °That root certificate signs a second certificate, often called the intermediate certificate
- °The intermediate certificate signs your custom certificate
- °As a result, you can trust the custom certificate.

World-wide, approximately 90% of the certificates are issued by the top four CA's, although that concentration is changing. Unfortunately, in recent years several

CA's have had their issuing authority revoked due to security deficiencies and violations. FileMaker Platform developers and FileMaker Server administrators should always consult the FileMaker, Inc. Tech Info System (Knowledge Base)³ for a listing of the approved CA vendors.

FileMaker, Inc. is very explicit about the approved type of the SSL certificate coming from each vendor. That there are different types of ownership validation, and that developers and administrators should carefully select the correct type certificate from the CA and avoid buying the *wrong type* from the *right vendor*.

The vendor sites (GoDaddy, Comodo, GeoTrust, Symantec, Thawte) offer different types of certificates. Those types include:

- Extended Validation SSL Certificates
- Organizational Validation SSL Certificates
- Subject Alternative Name (SAN) SSL Certificate
- Wildcard SSL Certificates
- Code Signing Certificates
- Unified Communications (UC) Certificates

As mentioned before, use the FileMaker Knowledge base to help pick the correct type of SSL certificate from your preferred vendor.

FileMaker Server does support both Wildcard and SAN certificates for the supported types listed. A wildcard certificate is typically issued in the form of *.company.com (with the "*" wildcard meaning "any number of subdomains for that domain," This means multiple FQDNs—fully qualified domain names—but all belonging to the same domain). An example wildcard certificate would cover:

- server1.company.com
- server2.company.com
- server3.company.com
- ... and more

A SAN certificate can cover multiple FQDNs too but belonging to different domains. And all of the covered FQDNs must be listed explicitly on the certificate. A SAN certificate does not cover an unlimited number of names like a wildcard does. One SAN certificate for instance could cover these three FQDNs:

- server.company1.com
- server.company2.com
- server.company3.com

³ As referenced above: <https://thefmkb.com/14176> or <https://support.filemaker.com/s/answer/view?anum=000025609> [new system]

The exact process of using the CSR and validating that you are the owner of the domain for which you want the certificate varies a lot from vendor to vendor, so rely on their support if that process does not seem to go well.

Most vendors ask for what type of server the certificate is. Pick the “other” option instead of the specific ones such as “Apache” or “IIS”.

The SSL certificate is valid until a certain date. Within that validity timeframe your SSL vendor typically will let you “rekey” your certificate without having to purchase a new one. A rekey basically is the process of generating a new CSR and having a new SSL certificate issued based on that new signing request. It may be needed if you decide to use a new domain name or subdomain name, or if you have misplaced your key files or when you think they could have been stolen. If you keep your certificate and its intermediate bundle, and the ServerKey.pem in a safe place, you can re-use them if you need to reinstall your FileMaker Server. There will be no need to have your SSL certificate re-issued or re-keyed for the purpose of reinstalling FileMaker Server.

IMPORTING A CUSTOM CERTIFICATE INTO FILEMAKER SERVER

In FileMaker Server 17, as you go through the process of installing the server executable, the installation process offers you the opportunity to do that import.

—Continues Next Page—

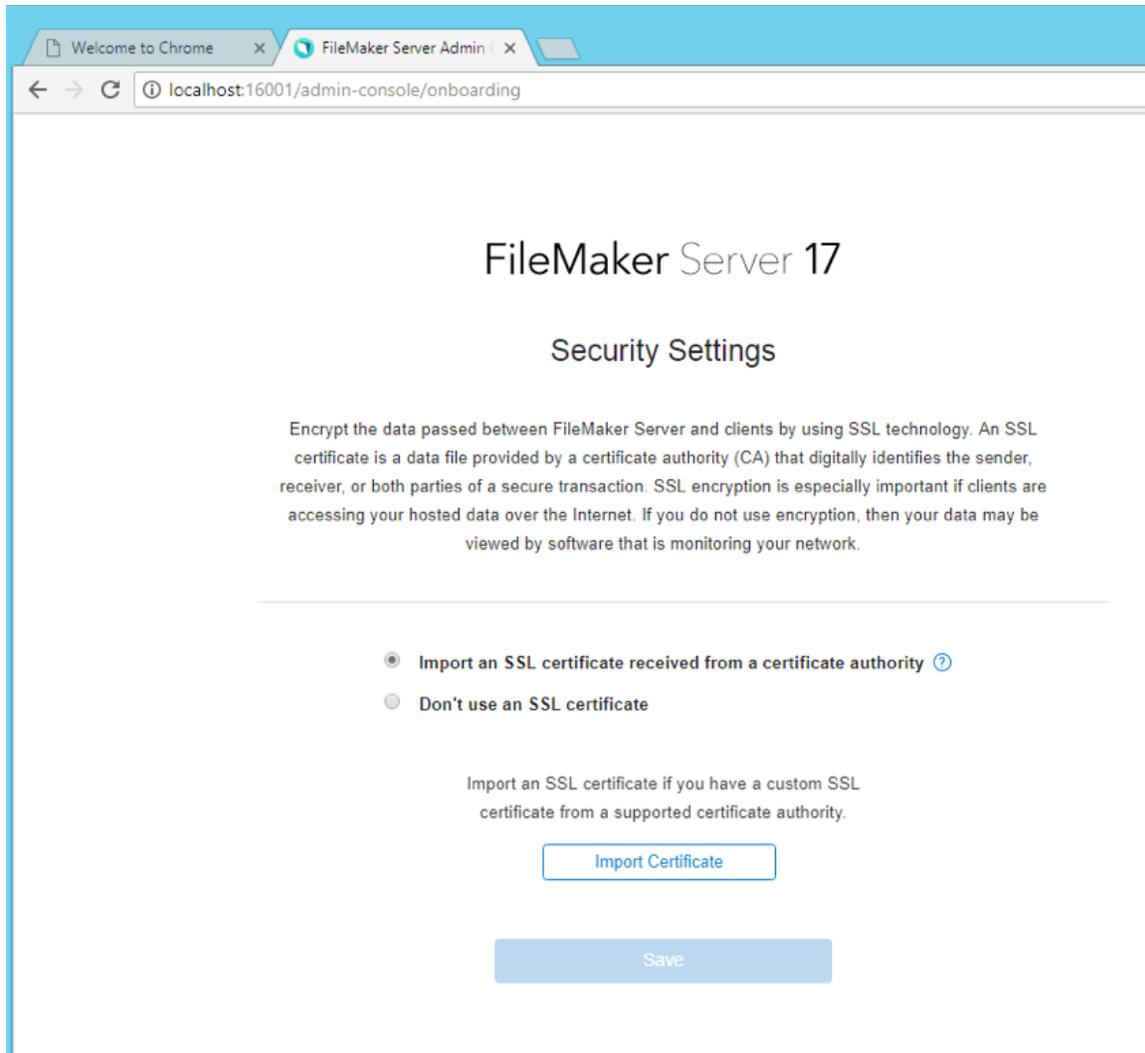


Figure 2. Importing SSL Certificate.

In *Figure 2* above, please note a new option: ***Don't Use SSL Certificate***. If selecting this option—and we recommend that you do not select it—the installer will present this dialog:

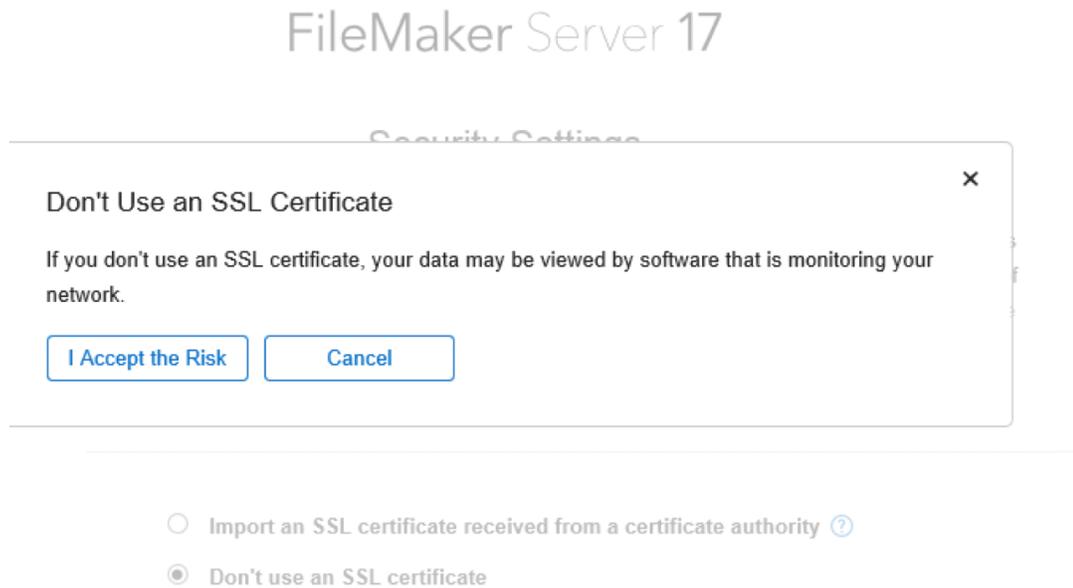


Figure 3. Accepting The Risk Of Not Using SSL.

FileMaker Server, as of version 17, still comes with a default SSL Certificate. The default certificate will be used automatically when you choose “Don’t use an SSL Certificate” above. But that default certificate will be used *only* for internal communication between the FileMaker Server components. It will not be used for Pro Advanced, Go, or WebDirect™ connections. **So unlike FileMaker Server 16 and earlier, you can no longer use that default certificate for those client connections. The only way to use SSL to protect the data in transit is to use a custom SSL certificate.**

When you are ready to import the Certificate for use with FileMaker Server, the User Interface of the new FileMaker Admin Console presents this dialog for your use:

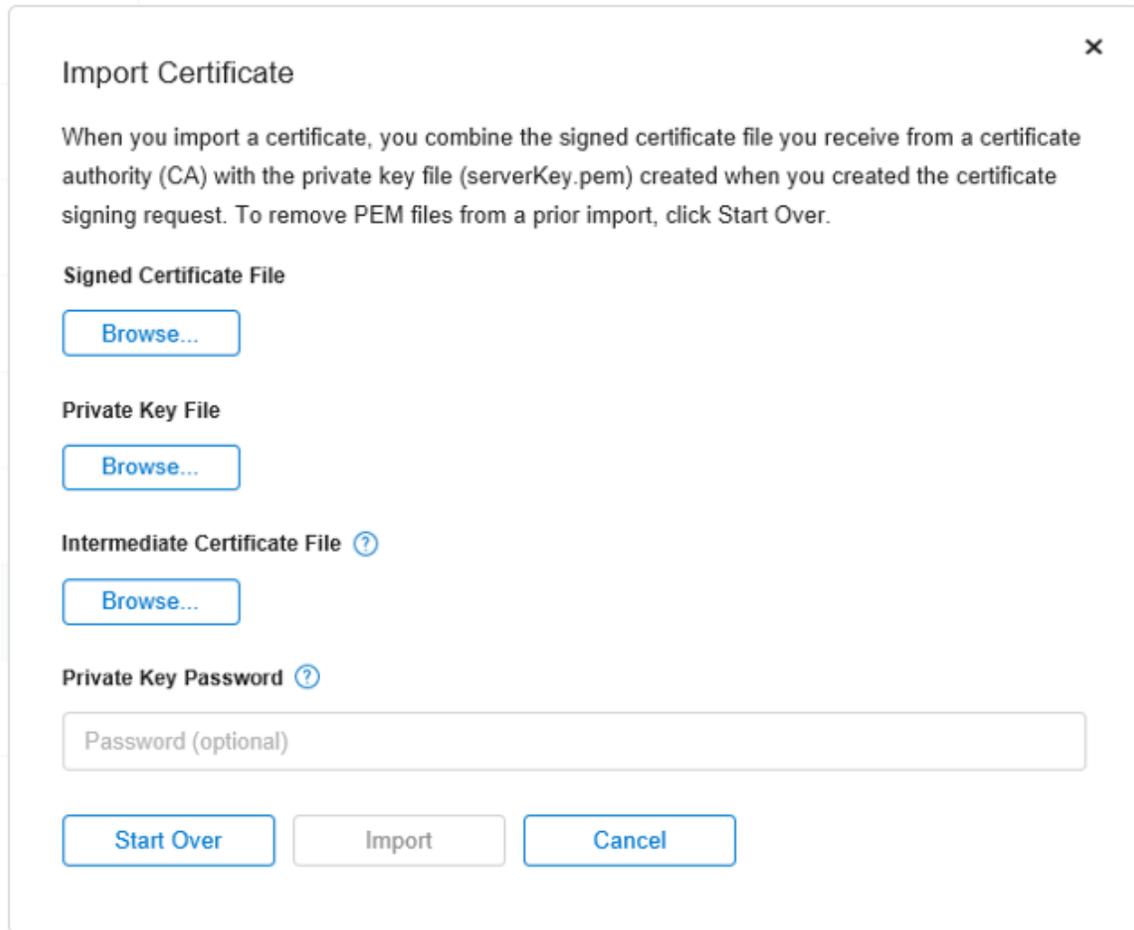


Figure 4. Selecting and Importing SSL Certificate.

Presuming a successful import of the SSL Certificate, the installer presents the following message:

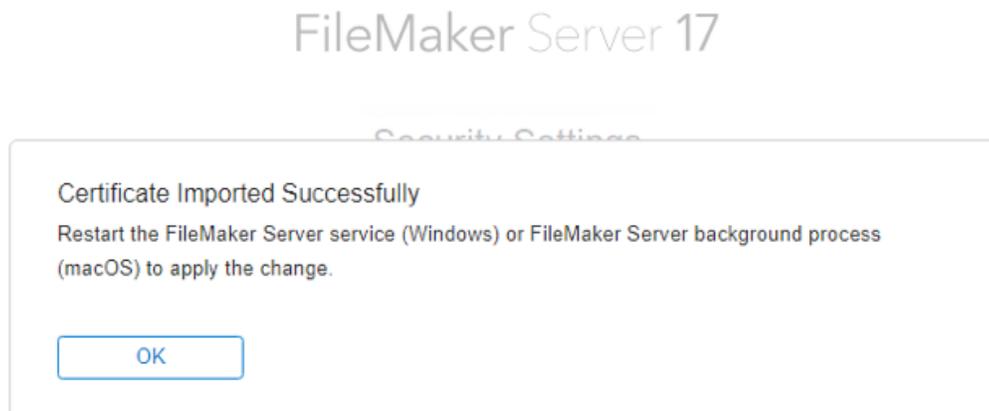


Figure 5. Successful Import Of Certificate.

A SIDE NOTE: HSTS CONNECTIVITY

HTTP Strict Transport Security (HSTS) is a security mechanism that requires any interaction with a web server always to use the HTTPS protocol instead of the insecure HTTP protocol. FileMaker® Server 16 allowed for use of this protocol to be optional even when using a custom SSL Certificate as shown below. This meant that if the user initiated an HTTP connection, it would be switched automatically to HTTPS when this setting was enabled. Since it could be turned off in FileMaker Server 16, connections to the server could be made through the insecure HTTP protocol even when a custom SSL certificate was installed, and SSL was enabled.



Figure 6. Earlier versions allowed HSTS to be optional.

In FileMaker Server 17, this has changed. HSTS will *always be invoked* whenever using a custom SSL Certificate. This means that insecure HTTP connections will not be allowed when you use a custom SSL certificate.

TESTING FOR SECURE CONNECTIONS TO FILEMAKER SERVER

Once the process completes, developers can check for a secure connection as follows:

- Connect to a file hosted on the Server by selecting the Server according to its *Fully Qualified Domain Name*. Using the Server IP address will not work correctly. This will likely return a Connection State of 2 with the orange lock icon.

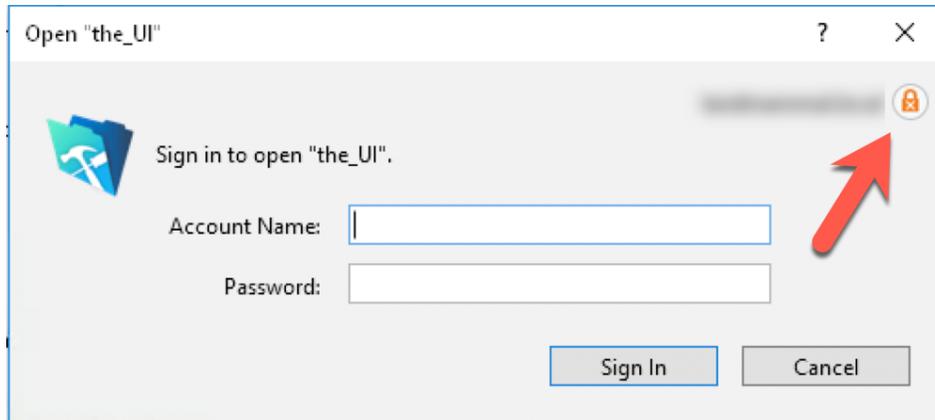


Figure 7. Orange Lock Associated With Connection State=2

- In the DataViewer or in a calculation invoke the **Get(ConnectionState)** function. It will return one of four different results based on the SSL situation in force.
 - **0** for no network connection for the current file (not applicable here).
 - **1** for a non-secured connection (FileMaker Server with SSL disabled, or to a FileMaker Pro Advanced host). This is equivalent to a red lock in the FileMaker user interface.
 - **2** for a secured connection (SSL) when the server name does not match the certificate (default FileMaker Server installation for older versions). This is equivalent to an orange lock.
 - **3** for a secured connection with a fully verified server name in the certificate. This is equivalent to a green lock.

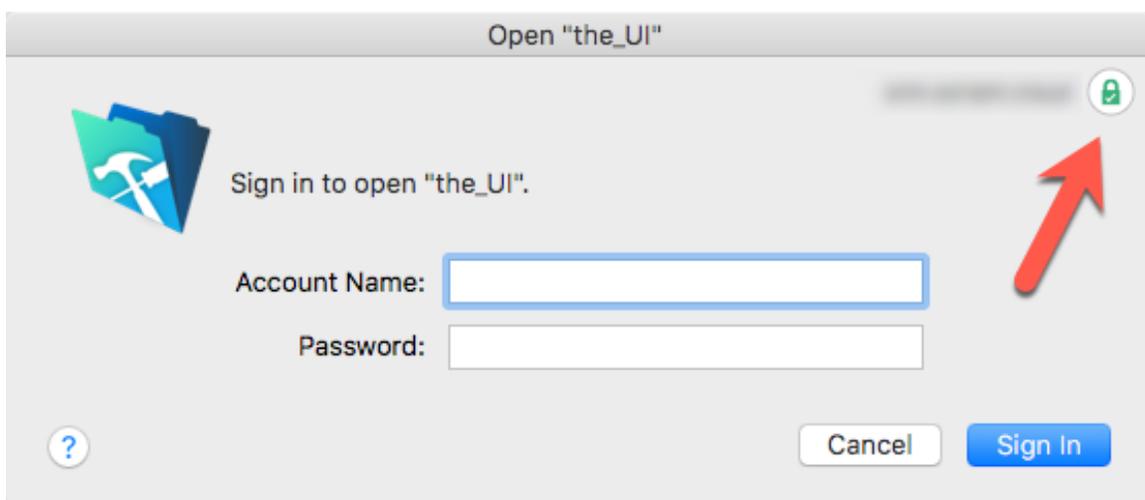


Figure 8. Green Lock Associated With Connection State=3.

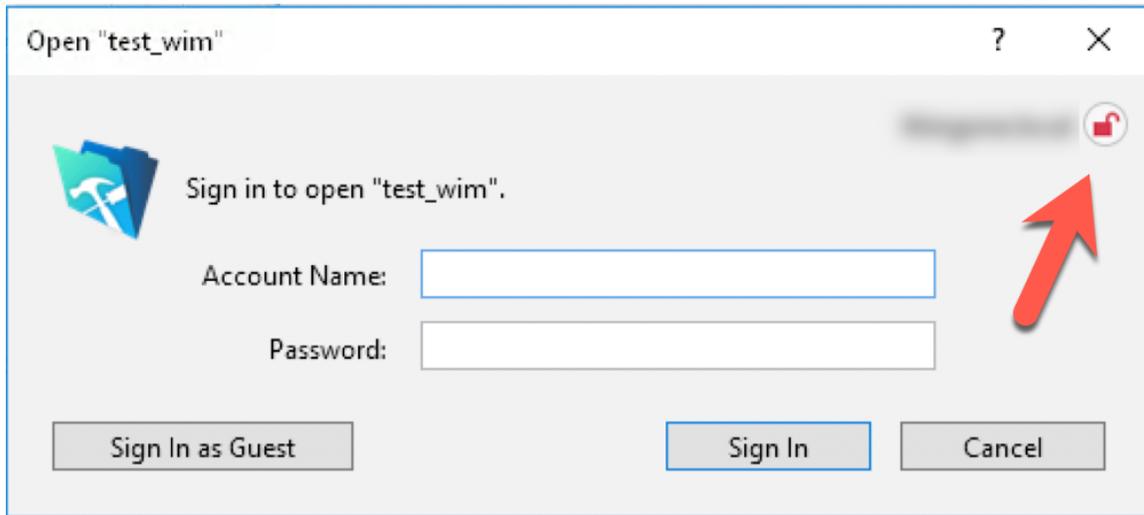


Figure 9. Red Lock Associated With Connection State=1.

Also, in the Dashboard of the new FileMaker Admin Console, you can see information about the expiration date of the SSL Certificate.

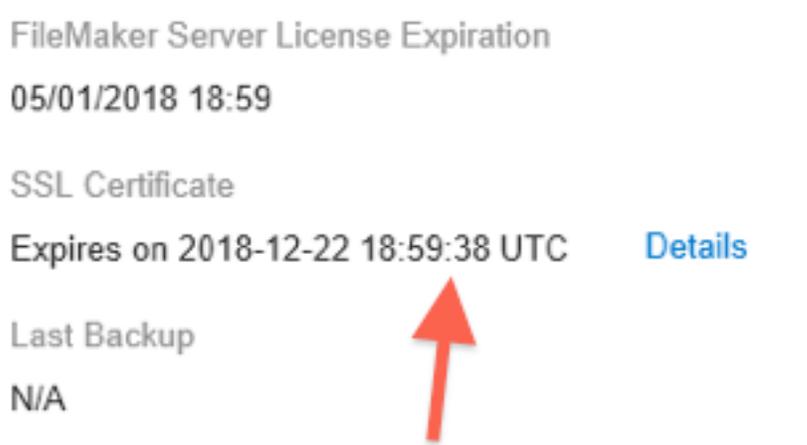


Figure 10. SSL Certificate Expiration Information

THE IMPORTANCE OF DNS

The statement made earlier that in order to achieve a “green lock” users must connect to FileMaker Server through the FQDN covered by the SSL certificate is crucial; and, it has an important prerequisite: that the server’s FQDN address can be resolved by the client’s workstation or device.

In practical terms that means that a working Domain Name System (DNS)⁴ has to be in place on the network so that when the user reaches out to server.company.com, that this request is routed to the correct server on the network. The server will recognize it is being *addressed by the name corresponding to its SSL certificate*, and the traffic between the client calling and the server will be properly encrypted.

If the user uses the FileMaker Server’s IP address for instance (say that it is 192.168.1.17), the user will be able to connect. However, when FileMaker Server responds with its FQDN that is on the certificate, the client will note that the server responds with a name (server.company.com) that is different than the name it called the server on (192.168.1.17). Therefore, the client will not “trust” that name provided by FileMaker Server, and will show an orange lock as a warning.

So, proper DNS is necessary to take advantage of the encryption in transit. While adding an entry to the client’s Operating System “hosts” file will also work in lieu of a DNS server, that is not a recommended method as it is brittle and not centrally managed as a proper DNS is.

One complicating factor in this can be Bonjour. Bonjour allows services (like FileMaker Server) to advertise itself to clients that are able to pick up on those advertisements without the need of a central DNS server. Bonjour involves networking concepts such as Zero Configuration⁵ and Multicast DNS (mDNS)⁶ all of which are beyond the scope of this document.

Bonjour uses the “.local” top level domain. So “server.company.com” would become “server.local” for a client seeing a Bonjour-advertised FileMaker Server. When the client connects to it, the SSL handshake is not complete. This is because the name “server.local” that the client used to address FileMaker Server does not match “server.company.com” that FileMaker Server will respond with as part of the SSL handshake.

In practical terms: a client will connect through Bonjour when:

- Bonjour is active on the FileMaker Server. (Bonjour is always installed on macOS. On Windows, the FileMaker Server installer will ask if you want to install it, but it will always use it if it is already installed.⁷)
- The user selects the server from the available local servers (as opposed to using a favorite entry that uses the proper FQDN) in the FileMaker Pro Advanced “Hosts” menu (“Open Remote” in FileMaker Pro 16 and earlier).

⁴ https://en.wikipedia.org/wiki/Domain_Name_System

⁵ https://en.wikipedia.org/wiki/Zero-configuration_networking

⁶ https://en.wikipedia.org/wiki/Multicast_DNS

⁷ To remove Bonjour after installing it, see <https://www.soliantconsulting.com/blog/filemaker-15-ssl>

•The FileMaker Pro Advanced client will show a warning as shown in *Figure 11* where you will see that the client is using the “.local” domain as part of the server’s name.

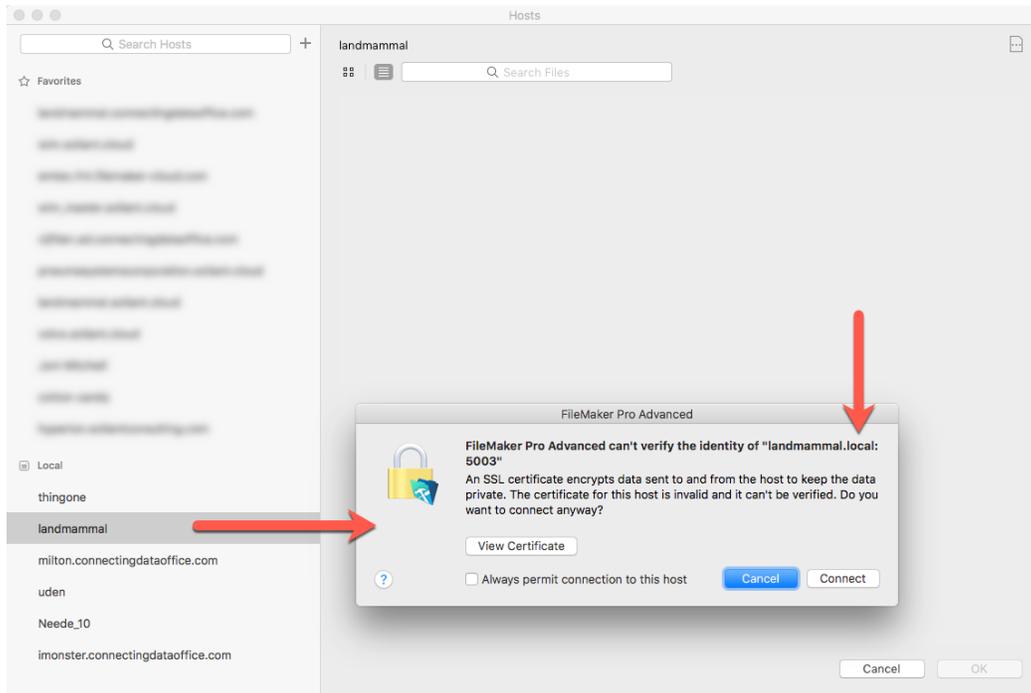


Figure 11. Warning For servername.local.

After selecting “Connect” and opening a file, the connection to that hosted file on that server will result in an orange lock. This is despite that server’s having a valid SSL certificate installed. With an “orange lock” the *data is still encrypted in transit, but the name validation was not successful.*

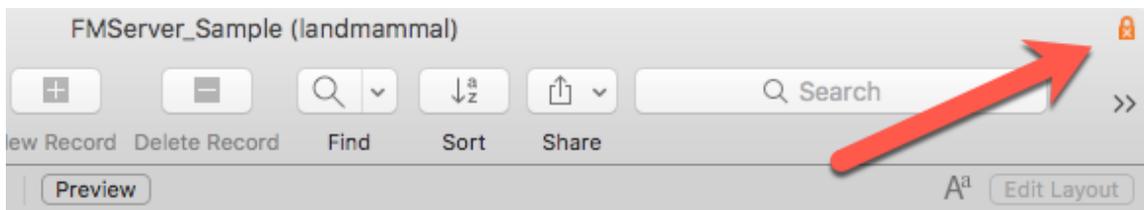


Figure 12. Orange Lock Connection Denotes Name Validation Unsuccessful.

When you create a favorite host using the **Fully Qualified Domain Name** for the server and connect to it, then there is no warning, and the connection results in a green lock as shown below:

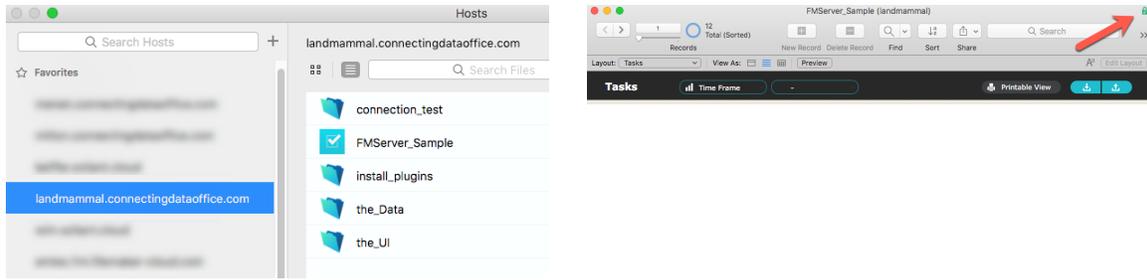


Figure 13. No Challenge When Selecting Favorite Host And Green Lock On The Connection.

In short: Bonjour can make it harder to achieve “green lock” status if your users rely on the “local” section of the Hosts dialog in FileMaker Pro Advanced and Go. To ensure a green lock, make sure the users are using a “favorite host” entry, or make sure that any references to the hosted file from a launcher file use the proper FQDN of the server.

MULTI-MACHINE DEPLOYMENTS

When your installation splits the FileMaker Server roles over different machines with multiple worker servers, then you need to import the certificate on each of those worker machines using the same command line. You do not need to generate a CSR for each machine if you use a wildcard or SAN certificate.

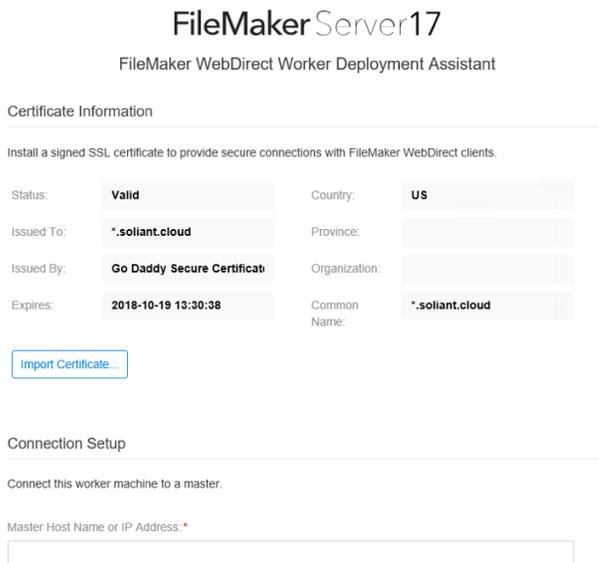


Figure 14: Import SSL Certificate On The Worker Machine Deployment Wizard.

Note that when connecting the Worker Machine to the Master Machine with a host name rather than an IP address, that has to be done through a Fully Qualified

Domain Name. And that implies that a custom SSL certificate is already installed on the Master machine.

Connection Setup

Connect this worker machine to a master.

Master Host Name or IP Address: *

Worker Host Name or IP Address: *

Admin User Name: *

Admin Password: *

When you enter a host name, use a Fully Qualified Domain Name (FQDN).



Figure 15. Connection Setup Of A WebDirect™ Worker Machine.

OTHER CLI OPTIONS

We previously described the “fmsadmin certificate” options to create a CSR and import the certificate. There is one more available option, **delete**.

To *remove* an imported certificate, use the CLI command:

```
fmsadmin certificate delete
```

and restart FileMaker Server for the change to take effect. You may have to use this if you are getting an error on importing the certificate, or if you need to change the SSL certificate to a different FQDN.

In the FileMaker Server Admin console, this is done by clicking the “Start Over” button on the SSL import dialog:

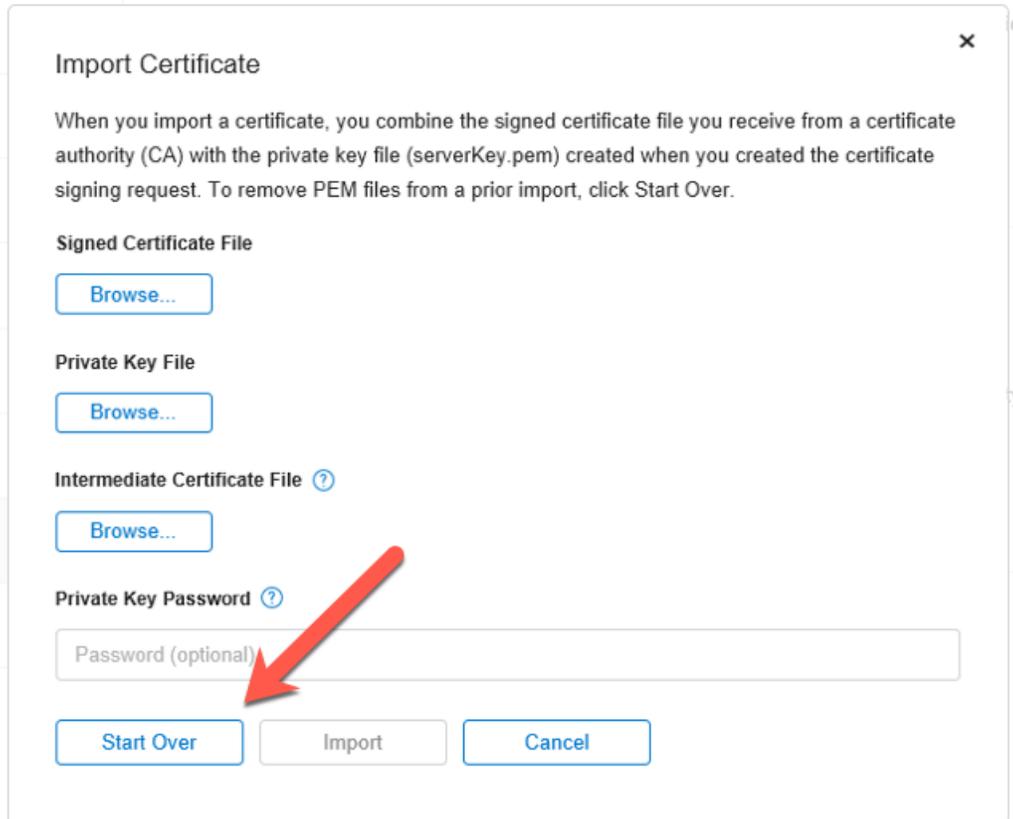


Figure 16. Deleting Certificate And Starting Over.

—Summary—

In this White Paper, we have described the nature of encryption and have discussed how it is used with FileMaker Server, both to protect data in transit across networks and to verify identity assertions made by FileMaker Server machines. We have also discussed how to acquire the necessary Custom SSL Certificates and how to install them.

In conclusion: as FileMaker, Inc. continues to improve the security features across all of the Platform, the use of SSL will become more and more predominant. Whether the data reside on FileMaker Cloud, on another cloud host provider, or on-premise, the general agreement is that encryption of the data in transit is a basic requirement.

—ACKNOWLEDGEMENTS—

The authors have drawn extensively on materials prepared by **Mislav Kos**, Technical Project Lead at Soliant Consulting, in conjunction with his 2017 FileMaker DevCon presentation.⁸ We appreciate very much Mislav's assistance.

The authors also very much appreciate and wish to acknowledge the assistance and support of a number of persons at FileMaker, Inc. in the preparation and review of this White Paper. They made it a better work. We and the entire FileMaker Developer Community are indebted to them:

- ❖ Rick Kalman
- ❖ Robert Holsey
- ❖ Sangita Banerjee
- ❖ Emmanuel Thangaraj
- ❖ Melody Hsu
- ❖ Lisa Rose
- ❖ Brian Maas

⁸ <https://www.youtube.com/watch?v=BvAVdbIH2ro>

—ABOUT THE AUTHORS—

WIM DECORTE is a Senior Technical Architect at Soliant Consulting, a FileMaker Business Alliance Platinum Member company. He is a leading expert on FileMaker Server, FileMaker Platform integration, and IT infrastructure issues. He is the author of numerous White Papers, Technical Briefs, and BLOG posts.

STEVEN H. BLACKWELL is a FileMaker Business Alliance Platinum Member Emeritus. He is the author of *FileMaker Security: The Book* as well as numerous White Papers and Technical Briefs about FileMaker Platform Security. He is also the creator of the FileMaker Security BLOG (<http://fmforums.com/blogs/blog/13-filemaker-security-blog>)