o

File & Serve

E-Filing Service Provider (EFSP) Specification

Odyssey®

Copyrights and Confidentiality

Publishing History

Document Publication Number	Date	Changes Made
ODY-FS-230-3188 v.1	June 2013	Document Creation
ODY-FS-230-3188 v.2	Jan 2015	Updates Business Process diagram and descriptions. Removed Appendices as API specifics are now covered in the interactive EFSP API Reference Guide
ODY-FS-230-3188 v.3	Sep 2015	Updated Payment Processing section to include e-Checks.
ODY-FS-230-3188 v.4	Dec 2015	Updated Figure 7 to include the XML namespace necessary for the UserNameHeader. Noted that HTTPS must be TLS 1.1 or TLS 1.2
ODY-FS-230-3188 v.5	May 2016	Updated to reflect support for service in general, not just electronic service. Updated Figure 3 to account for configurations that allow for more than one attorney per case party.
ODY-FS-230-3188 v.6	Oct 2016	Recommendation to vendors to utilize NTP system clock synchronization
ODY-FS-230-3188 v.7	Feb 2017	Updated the API Business Process diagram and the description of step #5 to abstract away the two CMS callback models.
ODY-FS-230-3188 v.8	Sep 2017	Stated that interactions with TOGA are via HTTP POST. Noted that the EFM supports two methods of transmitting documents to the EFSP.
ODY-FS-230-3188 v.9	Feb 2022	Support for ECF 5 Requirement for TLS 1.2 Support for TLS 1.3 coming later in 2022
ODY-FS-230-3188 v.10	Sep 2022	Reference to additional specifications (re:Search CMS, User Authentication, Composable Security, In-App Notifications, and User Preferences) TLS 1.3 support, pending upgrade of underlying systems. Added a reference to the User Authentication Specification in the Message Security section Added a reference to the Composable Security Model Specification in the Firm Administrator Setup Tasks section

Technical Assistance

For assistance, contact Tyler Technologies through the following resource numbers:

Main (Reception) Telephone	972.713.3770
Main (Reception) Fax	972.713.3777
Courts & Justice Client Support Phone	800.966.6999 and 877.874.8499
Courts & Justice Client Support Fax	972.713.3780
Courts & Justice Client Support Website	tylertech.com/client-support/courts-and-justice-support
Courts & Justice Technical Publications	planodocfeedback@tylertech.com
tyler community	tylercommunity.tylertech.com

Contents

List of Figures

About this Document

The *Odyssey® File & Serve EFSP Specification* contains the information needed to integrate with the E-Filing Manager (EFM) —the central hub in the File & Serve e-filing solution.

Odyssey File & Serve Documentation

The File & Serve e-filing documents available include*:*

- *Odyssey® File & Serve Solution Overview*
- *Case Management System (CMS) Specification*
- *Case Management System (CMS) API Reference Guide*
- *re:Search Case Management System (CMS) Specification*
- *E-Filing Service Provider (EFSP) Specification*
- *E-Filing Service Provider (EFSP) API Reference Guide*
- *User Authentication Specification*
- *Composable Security Model Specification*
- *In-App Notification Specification*
- *User Preferences Specification*

Documentation Conventions

This table contains descriptions of the text conventions used in File & Serve technical documents.

Text Style	Description	Example
Underline blue	URLs, links to another section in same document or external document	www.tylertech.com
Bold	Menu, menu item, menu, tab, button, dialog box, element/field, heading, page, pane, panel, section, program names. Options in drop-down lists, system and application messages	**Tools** > **Options** > **Forms** **Instances** tab
`Fixed-Width`	Server names, filenames, pathnames, databases, API names, XML settings, scripts, command names, system calls, data structures and types	Run the `tables.sql` script for the `jcpBasketB` database
`Fixed-Width` **`bold`**/**`blue`**	Command line: Input, operators, code samples	`D:\ProgramFiles\neevia.com\docuPrinterSDK\.Net`

Bold **Fixed-Width**	Input typed exactly as shown.	Set odysseycom as the domain user. Run the script `DROP TABLE`.
Italics	Placeholders in syntax Document titles and emphasis	Type the *filename*. *Deployment Troubleshooting Guide*
Quotation marks	Chapter titles	Refer to Chapter 3, "Server Settings."

E-Filing Terms

This table contains descriptions for the terms used in Odyssey File & Serve documentation.

Term	Description
Attachment	Series of bytes in the message stream, after messages, transmitted between MDEs that constitutes, in whole or in part, an electronic document whose conventional equivalent would be a document on paper.
Case Management System (CMS)	Electronic record of court information and documents.
Docketing	Process invoked/triggered when a court receives a plea, order, or notice (error-free transmission and presentation of required content) and records it as a part of the official record.
Document	Electronic equivalent of a hard-copy document.
ECF Architecture Specifications	Core specification: Defines the MDEs and operations and messages exchanged between the MDEs. Service Interaction Profiles: Specifications that describe the communication infrastructure that delivers messages between MDEs. Document Signature Profiles: Specifications that describe the mechanisms for signing electronic documents.
E-Filing Manager (EFM)	Central hub for e-filing operations: Receives e-filing submissions from EFSP, provides a review process for courts, and transmits e-filing data to the appropriate CMS.
E-Filing Service Provider (EFSP)	Primary interface to a filer: Interacts with an EFM to perform e-filing operations.
Filer	Attorney or a pro se (self-represented) litigant acting as an individual who assembles and submits one or more filings (combinations of data and documents).
Filing	Electronic document (with associated data and attachments) assembled for filing into a specified court case.
Major Design Element (MDE)	Logical grouping of operations (`Court Policy MDE`, `Filing Assembly MDE`, `Filing Review MDE`, `Court Record MDE`, `Service MDE`, and `Court Scheduling MDE`) representing a significant business process supported by ECF. Each MDE operation receives one or more messages, returning a synchronous response message (a reaction to a message received) and optionally returning an asynchronous response message later to the originating message sender.

Term	Description
Message	Well-formed XML data structure transmitted between MDEs that can include one or more attachments.
Message transmission	Sending one or more messages and associated attachments to an MDE. Each transmission must invoke or respond to an operation on the receiving MDE, as defined in the ECF specification.
Method	Callable unit (subroutine) that performs a specific task.
Operation	Function, provided by an MDE upon receipt of one or more messages, represents a significant step in the court filing business process. Filer invokes by transmitting a request with an operation identifier and a set of messages.
Operation signature	Definition of the input message and synchronous response message associated with an operation. Operation gives message a name and type, where type is defined by one of the message structures defined in the ECF specification.
Web Service	Provides a set of functions, where a function is one specific task/operation.

The Odyssey® File & Serve Solution

Odyssey File & Serve, Tyler's e-filing solution, is a platform that serves as both an EFSP and an EFM to achieve interoperability with court case management systems (CMSs). It is based on the *Electronic Court Filing Specification 5.0* (ECF) to standardize data, transport methods, and the process of electronic filing between systems.

ECF Interoperability Compliance

E-filing service providers and different courts implement common filing behaviors and specifications for how data and messages are structured, transmitted, and received. Implementations of Odyssey File & Serve specifications utilize XML to create and transmit legal documents to and from courts and between appropriate parties. Three major elements of the ECF architecture set the requirements for EFSP and court compliance. These elements are:

- Major Design Elements (MDEs): Six MDEs support the operations and messages involved in the e-filing process.
- Operations: Grouping of logical operations into a single MDE to execute a business operation such as create a filing, or receive and record a filing.
- Messages: Well-formed XML data structure transmitted between MDEs that include one or more attachments.

New Functionality

The EFM operates within many jurisdictions, therefore the use of new functionality is often dependent upon jurisdictional requirements and any associated configuration. Jurisdictional requirements are not represented within this documentation.

Tyler strives to avoid "breaking changes" to the integration APIs (e.g. changing/moving existing schema elements). This includes using existing ECF elements wherever possible, and appropriate per ECF standards. However, the addition of schema is not considered a breaking change. Finally, if any breaking changes are necessary to accommodate e-Filing requirements, we work diligently to minimize the impact to our integration partners.

The entire EFM documentation set include additional details as necessary. As always, rely upon the EFSP Guide and CMS Guide to lead you to relevant artifacts for each API.

Electronic Filing API Business Process

The figure in this section illustrates the primary business process operations—case identification, filing assembly and submission, clerk review, post-review processing & docketing, and notification —of an electronic filing envelope. An envelope can contain one or more filings, but all filings in an envelope must pertain to one and only one court case.

Note: This figure does not cover all available APIs – refer to the EFSP Guide for full coverage.

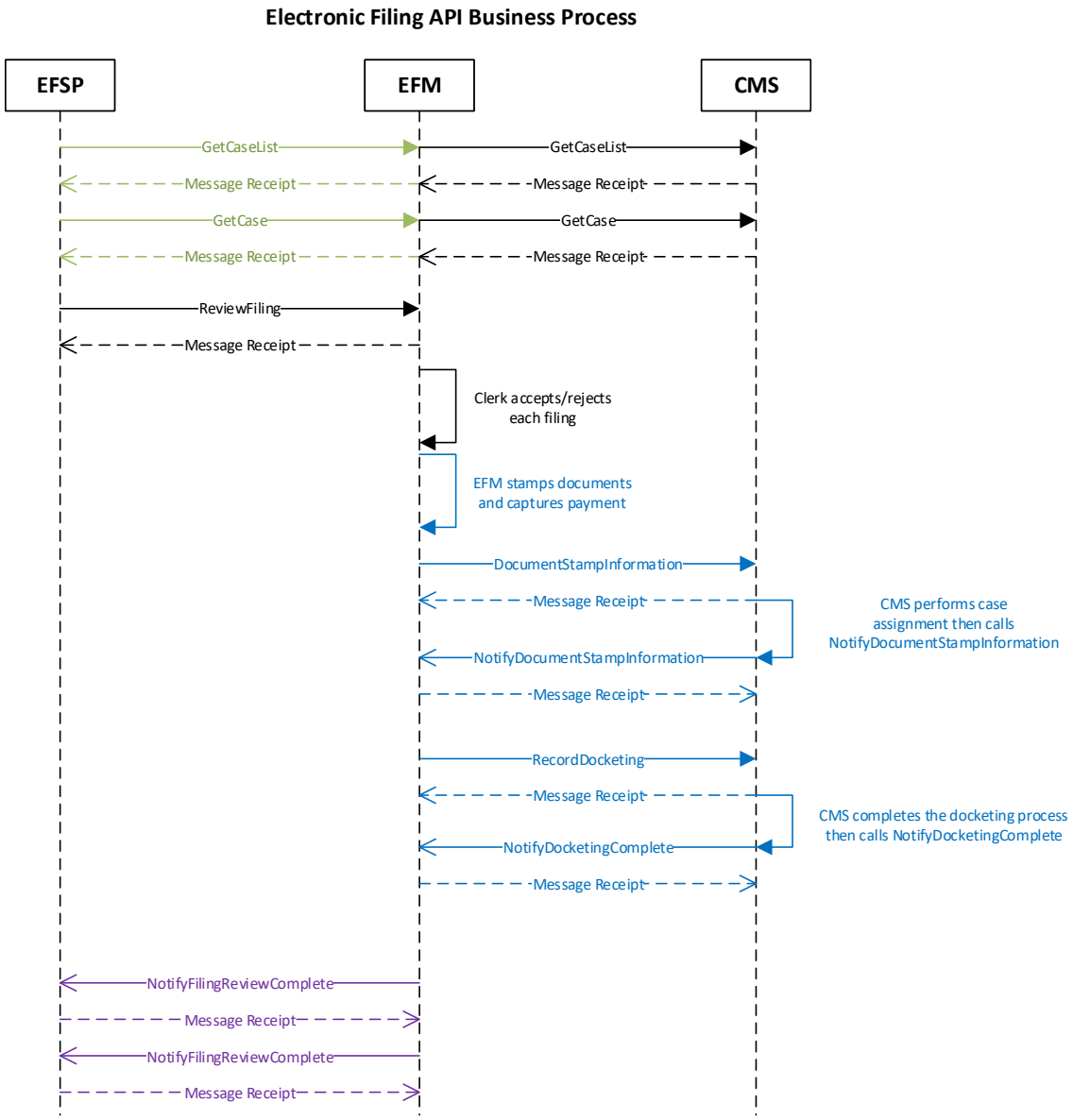

Figure 2 - Electronic Filing API Business Process

Additional details regarding each business process operation are provided below.

Process #1: Case Identification

The first step in the e-filing process is to identify the court case the filer wishes to file into. If the filer is filing a new suit, also referred to as an initial filing, then this activity is moot. However in order to file into an existing case, referred to as a subsequent filing, we must first identify the appropriate case. Two APIs provide for this activity:

- GetCaseList - performs a search operation
- GetCase - retrieves additional details about a specific case

The EFSP initiates these calls to the EFM. If the court's CMS is integrated to the EFM, the EFM will proxy the EFSP's calls to the CMS in order to provide real-time data; otherwise, the EFM will service the call itself based upon any information it has on hand based upon previous activity for the case.

Note: For non-integrated courts, filers can still file into pre-existing cases that the EFM has not processed before, but the filer must provide full case information and the court clerk will need to validate such information as part of the review process. Such cases that are unknown to the EFM are referred to as "non-indexed cases". Once the clerk validates the information within the EFM, the case becomes "indexed" and can thereafter be found and accessed by future EFSP calls to both the GetCaseList and GetCase APIs.

Process #2: Filing Assembly and Submission

Once the appropriate case has been identified, an envelope can be assembled and submitted to the EFM via the ReviewFiling API. The system uses a number of codified fields that are configured per court requirements, therefore the case's court location – the court or office in which the case resides – is important during the assembly process, in order to obtain the appropriate code values.

Each envelope is accompanied by a payment account, which designates the method of payment for the filing(s). Payment accounts are typed, so if the court has agreed to waive fees, the filer can submit the envelope with a waiver payment account and the clerk will accept or reject the filing on its own merit. If the payment account indicates payment by credit card, the EFM will perform what is known as an Authorize transaction through the payment processing system in order to determine whether sufficient credit exists on the credit card.

Process #3: Clerk Review

Once an envelope has been successfully submitted into the EFM for clerk review, the clerk will review each filing in the envelope and determine whether to accept or reject each filing. The Clerk Review process takes place within the Odyssey File & Serve application. The envelope and all of its filings remain in this business process until each filing in the envelope has been either accepted or rejected.

Process #4: Post Review Processing & Docketing

Once every filing in an envelope has been designated as either accepted or rejected, the envelope moves into the post review processing and docketing process. If all of the fillings were rejected, this process performs no actions. However, if one or more filings were accepted, the EFM will perform payment processing and

document stamping. If a CMS integration exists for the court, the EFM will initiate an exchange with the court's CMS to docket the accepted filings into the CMS. If no CMS integration exists, docketing will be performed manually by the court clerks.

Process #5: Notification

In order to complete the filing process, several notifications must take place.

An integrated CMS will:

1. If the CMS supports the DocumentStampInformation API, it will create the case in the CMS and then call the NotifyDocumentStampInformation API to pass case assignment information to the EFM.
2. Call the NotifyDocketingComplete API to notify the EFM that docketing within the CMS is complete.
    - NOTE: There are two callback models available to CMS vendors; However, this has no direct effect upon EFSPs.

The EFM will:

1. For each filing in the envelope, call the NotifyFilingReviewComplete API to notify the EFSP of the outcome of each filing.

EFSP Integration to the EFM

The E-Filing Service Provider (EFSP) is responsible for acting as the `Filing Assembly MDE` and generating the XML core filing message for submission to the court as an electronic filing. The central hub in the File & Serve e-filing solution is the EFM, as shown in this illustration.

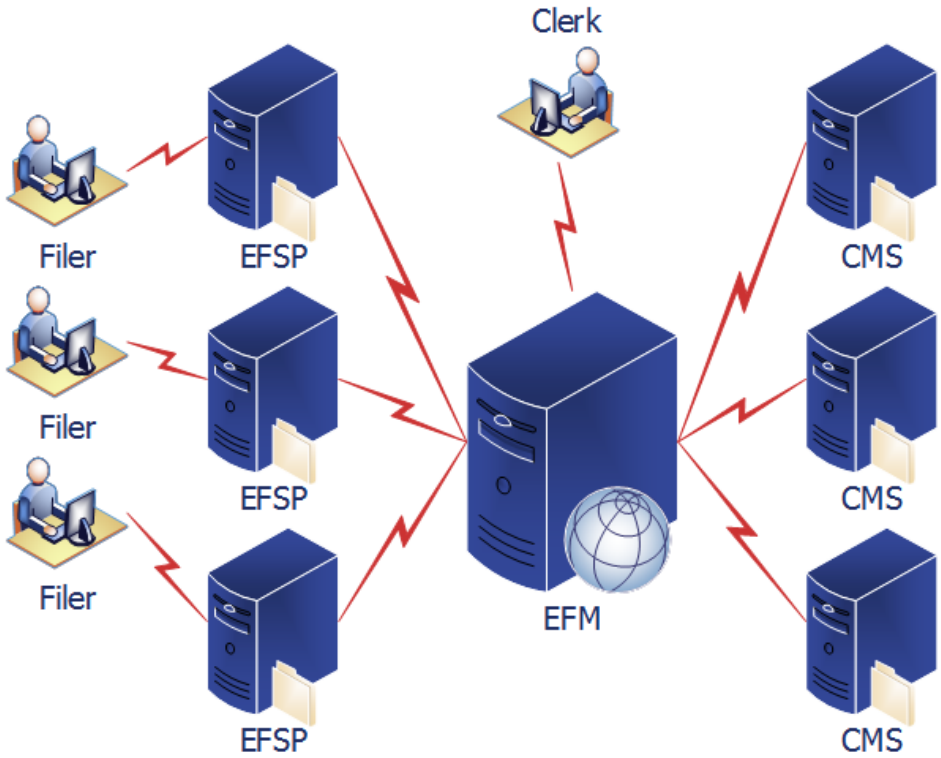

Figure 1 - File & Serve Topology

Administration

The base entity in e-filing is the *firm*. Even an individual user account has an underlying firm record. All operations revolve around a firm, which represents an organization that intends to share filing information. A firm contains user profiles, attorneys, service contacts, and payment accounts. File & Serve enables each firm to designate one or more persons as *firm administrator* of its e-filing system. A firm administrator registers the firm, sets up and maintains all user, payment, and attorney accounts.

The Firm

This diagram illustrates the relationships between the firm and its child entities.

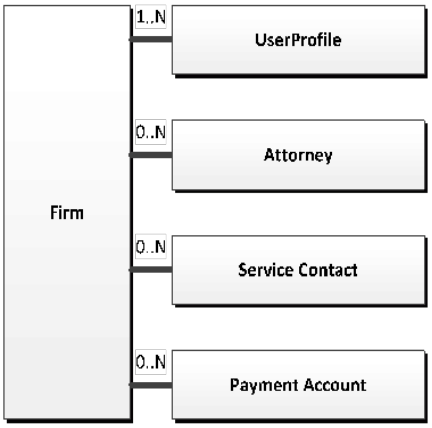

Figure 2 - Firm Entity Relationship Diagram

This table contains a description for each child entity.

Child Entity	Description
UserProfile	Represents a registered user. A firm must have at least one user. Individual user accounts may only have one user.
Attorney	Represents an attorney. While technically a firm can exist without an attorney, an attorney will be required in order to submit filings. Individual user accounts will not have an attorney.
Service Contact	Represents the contact information designating where a particular party receives service. Service contacts are required in order to receive service.
Payment Account	Represents a registered credit card or another payment mechanism.

File & Serve provides these functions for EFSPs in addition to the ECF supported EFM operations:

- User Management: The management of user accounts within the File & Serve system.
- Firm Management: The relationship of users within a firm, ability to manage the firm, identification of attorneys, and management of service contacts.
- Court Configuration: The ability to access court-specific configuration values.
- Payment Account: The management of payment accounts, which facilitates interaction with the payment processing system.

Firm Administrator Setup Tasks

This section contains an outline of the required tasks that a firm administrator must perform.

NOTE: The EFM 2022.1 release now supports the Composable Security Model (CSM) for user/role assignments. While the order of operations below is valid, some of the specifics vary when the CSM is enabled. Refer to the *Composable Security Model Specification* for more information.

1. Create the first user–firm administrator; the firm is created in conjunction with this user. Next, activate the firm administrator account.
   - To create the first user–firm administrator: Call `RegisterUser` with `type = FirmAdministrator`.
   - To activate the firm administrator account (required): File & Serve (EFM) sends an email message to the email address of the user. This user must click the link provided in the message in order to activate the account. Once activated, the user can sign on to the system and perform administrative tasks for the firm.
2. Create additional user accounts.

   The firm administrator can now add additional firm users to the system. These users are associated with the firm and granted the role of `Filer`.
   - To add a user: Call `RegisterUser` with `type = FirmAdminNewMember`.
   - To grant a user the `FirmAdmin` role: Call `AddUserRole`.
   - To activate a user account (required): File & Serve sends an email to the email address of the user. This user must click the link provided in the email in order to activate the account. Once activated, the user can sign on and use the system.
3. Create attorney records.

   The firm administrator should now create one or more attorney records for the firm. When created, these attorney records are associated with the firm.
   - To add an attorney: Call `CreateAttorney`.
4. Create service contact.

   The firm administrator should now create one or more service contacts for the firm. Each service contact represents a person associated with the firm who can later be attached to the firm's cases and receive service related to those cases. These created service contacts are associated to the firm.
   - To create a service contact: Call `CreateServiceContact`.
5. Create payment accounts.

   The firm administrator should now create one or more payment accounts—one for each credit card intended for use in the payment of filings. Additionally, the firm administrator can create a waiver payment account for filings where the court has waived fees.
   - To create a payment account: Call `CreatePaymentAccount`.

Filing

The EFM provides the ability to submit multiple filings for a case within a single transaction. A filing envelope represents filings grouped for a particular case; it is the File & Serve top-level element for the `ReviewFiling` and `RecordDocketing` operations.

Some key facts to note are:

- Each filing transaction sent to/from the EFM is represented by an envelope
- An envelope pertains to one and *only* one case
- Each case has one or more case parties
- Each case party has zero or one case party attorney
- Each envelope has one or more filings for the case

- Each filing has one or more documents
- Each document has one and only one filing component
- Each filing has zero, one, or more requests for service
- Each request for service has one and only one service contact to be served

The Filing Envelope

An envelope contains a case, case parties, case party attorneys, filings, documents, filing components, and optional services. Service is initiated at the envelope level; however, it is performed and tracked at the filing level. Each service has a corresponding service contact.

This diagram illustrates the relationships of the filing envelope and its child entities.

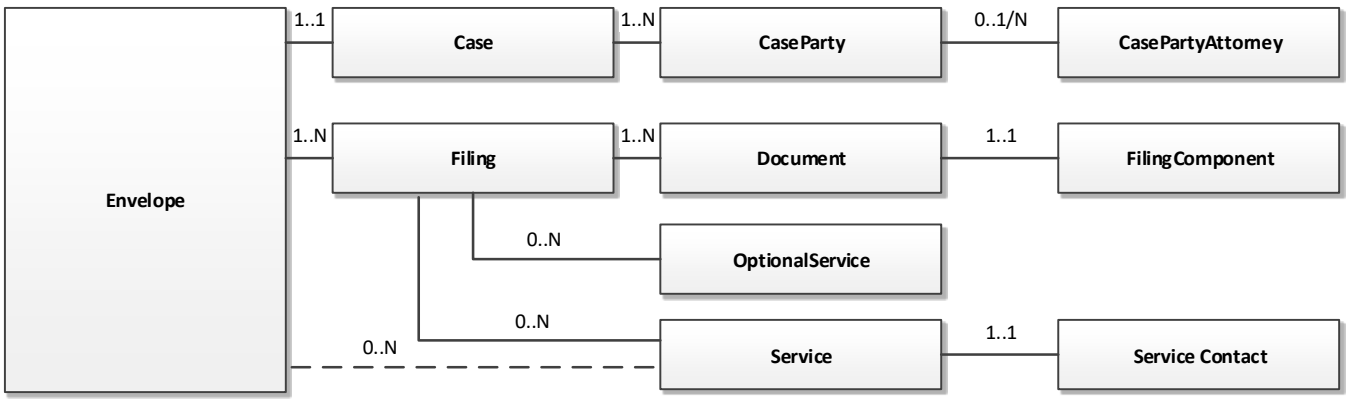

Figure 3 - Filing Envelope Entity Relationship Diagram

This table contains a description for each child entity.

Child Entity	Description
Case	Represents either an existing court case or a new court case being initiated.
CaseParty	Represents a party on a court case.
CasePartyAttorney	Represents an attorney representing a party on a court case. System configuration determines whether more than one attorney is allowed per case party.
Filing	Represents the metadata associated with a docket entry in a court case.
Filing Component	Filing Components are configurable features associated with individual filings. These components consist of types of documents needed to satisfy a filing. For example, a particular filing configuration requires a lead document and optionally allows for the addition of attachments (supplemental supporting documents). Each filing within an envelope must include at least one filing component for documents.

Child Entity	Description
Document	Legal document from a filer to a court or another authorized party.
Optional Service	Optional services are services offered by the clerk. The two most common examples are Service of Process and Certified Copies. The clerk often charges a fee for optional services and the EFM will collect any such fees as part of the filing.
Service	Instructs the EFM to perform service for a service contact. Electronic service will take place either before or after clerk review, depending upon court location specific configuration.
ServiceContact	Represents a person to receive service.

Network and Security Infrastructure

All system interactions between an EFSP and the EFM take place through secure web services using SOAP messages. These security requirements apply:

- Transport security: Messages exchange is over HTTPS. This ensures encryption of the HTTP request that can only be decrypted by the intended receiver.
- HTTPS traffic must utilize TLS 1.2. The 2022.1 release brings support for TLS 1.3; however, this may also require upgrades of underlying systems. Tyler will coordinate TLS 1.3 rollouts within each market.
- Message security: Messages contain a digital signature created in compliance with the *WS-I Basic Security Profile 1.1 Specification* using the EFSP private key. This ensures message integrity and asserts the authenticity of the EFSP.
- Tyler recommends that partner systems leverage NTP to synchronize system clocks to avoid message exchange failures.

ECF Web Services Service Interaction Profile

The *Web Services Service Interaction Profile 2.1 Specification* (SOAP messages over HTTP/HTTPS) defines the transmission system using the specifications described in the *Web Services Interoperability (WS-I) Basic Profile 1.1* and the *WS-I Basic Security Profile 1.1*. To employ the web services profile requires the availability of the appropriate web services for each MDE in order to initiate each of the required operations and submit messages for consumption.

SOAP Message Package

This figure illustrates the containment of ECF messages and attachments within a SOAP Message Package—the SOAP envelope (an XML document), one or more Multipurpose Internet Mail Extensions (MIME) parts containing binary octets.

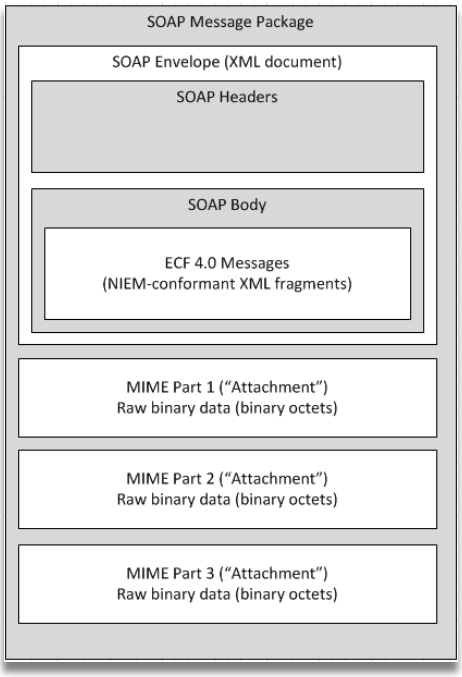

Figure 4 - SOAP Message Package

Example of Message Transmission Optimization Mechanism (MTOM)

All message exchanges are SOAP 1.1 packages that use the Message Transmission Optimization Mechanism (MTOM).

```
POST https://server/path/webservice.svc HTTP/1.1
MIME-Version: 1.0
Content-Type: multipart/related;
type="application/xop+xml";start="Envelope";start-
info="text/xml"
Host: server

--boundary
Content-ID: Envelope
Content-Transfer-Encoding: 8bit
Content-Type: application/xop+xml;charset=utf-
8;type="text/xml"

<s:Envelope>
  <s:Header>
    <o:Security>
      <u:Timestamp>
        ...
      </u:Timestamp>
      <o:BinarySecurityToken>
        ...
      </o:BinarySecurityToken>
      <ds:Signature>
        ...
      </ds:Signature>
    </o:Security>
```

```
  </s:Header>
  <s:Body>
    <ReviewFiling>
      <CoreFilingMessage>
        ...
        <DocumentAttachment>
          <BinaryUTF8Object>
            <xop:Include
href="cid:ElectronicDocument1" />
          </BinaryUTF8Object>
          ...
        </DocumentAttachment>
        ...
      </CoreFilingMessage>
      <PaymentMessage>
        ...
      </PaymentMessage>
    </ReviewFiling>
  </s:Body>
</s:Envelope>

--boundary
Content-ID: ElectronicDocument1
Content-Transfer-Encoding: binary
Content-Type: application/octet-stream

... binary data of the electronic document ...

--boundary--
```

Figure 5 - Message Transmission – MTOM Attached Document Example

Transmission of Documents

The EFM supports two models for transmitting documents to the ESFP for the NotifyFilingReviewComplete API:

1. Embedded within the SOAP XML (base 64 encoded)
2. Sending a URL by which the documents can be downloaded

The model to use is determined via configuration within the EFM for each EFSP.

Message Security

Operations that require an authenticated user expect the user credentials in the form of SOAP message headers. The user password is only sent in clear text in the SOAP message body when the password itself is part of the operation, such as the `AuthenticateUser`, `RegisterUser`, `ChangePassword`, and `ResetUserPassword` operations. These operations return a hash of the password to the calling EFSP. Subsequent web service calls representing the authenticated user should include the user name and hashed password in the SOAP message headers.

NOTE: EFM 2022.1 now supports the use of Okta for user authentication. Refer to the *User Authentication Specification* for important updates regarding message security.

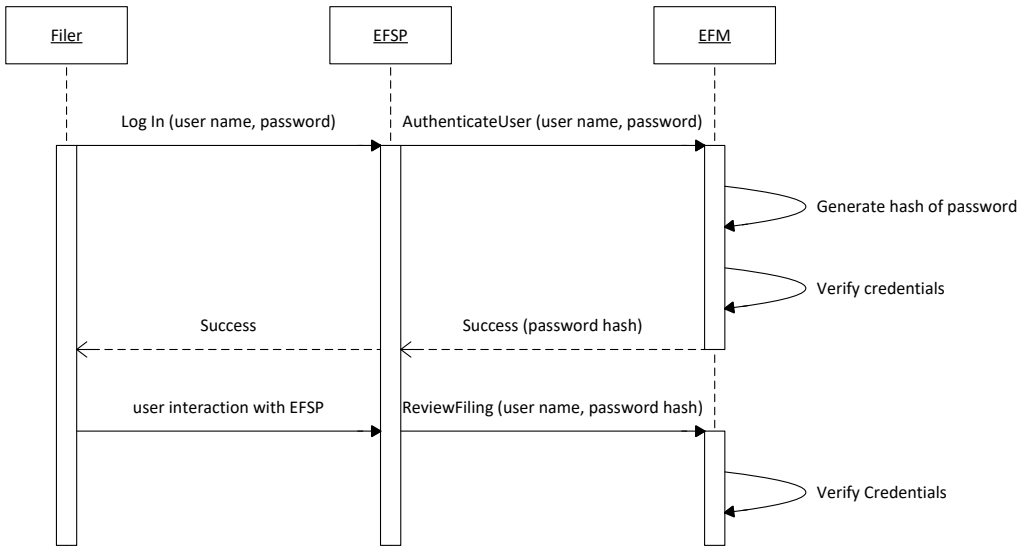

Figure 6 - Hashed Password

This is an example of a SOAP message with user credentials within the header.

```
<?xml version="1.0"?>
<s:Envelope>
  <s:Header>
    <UserNameHeader xmlns="urn:tyler:efm:services">
      <UserName>(user name)</UserName>
      <Password>(hashed password)</Password>
    </UserNameHeader>
    <o:Security>
      <o:BinarySecurityToken>(binary
```

```
data)</o:BinarySecurityToken>
      <Signature>
        …
      </Signature>
    </o:Security>
  </s:Header>
  <s:Body>
    …
  </s:Body>
</s:Envelope>
```

Figure 7 - SOAP Message Example

Error Handling

Web service calls initiated by the EFM include the use of an automatic retry mechanism in the event of a timeout or failure. Once this retry mechanism expires, the message routes to an exception queue for manual intervention.

Error Identification

In general, the EFM attempts to identify all applicable errors in one pass rather than stopping after a single error. However, the ability to follow this pattern depends on the severity of the error encountered. The delivery of a complete list of errors produced by the EFM is scheduled in conjunction with the test system.

Payment Processing

This chapter provides developers with guidance in enlisting the Tyler Online Gateway (TOGA) to collect account information from filers. Payment processing is performed through the Tyler Online Gateway (TOGA) in partnership with Chase Paymentech. TOGA registers credit card and/or eCheck information with the Chase Paymentech system for exclusive use with File & Serve. When Chase registers the account, it issues a payment token. This relieves the EFSP from collecting and/or storing sensitive credit card information.

Payment Process

Upon receipt of a payment token from Chase, an EFSP sends a request containing the token to File & Serve to create a payment account within the EFM. The process flow diagram illustrates the interaction between the EFSP portal and TOGA. Sample user interface pages for the Tyler Online Gateway process functions follow the diagram.

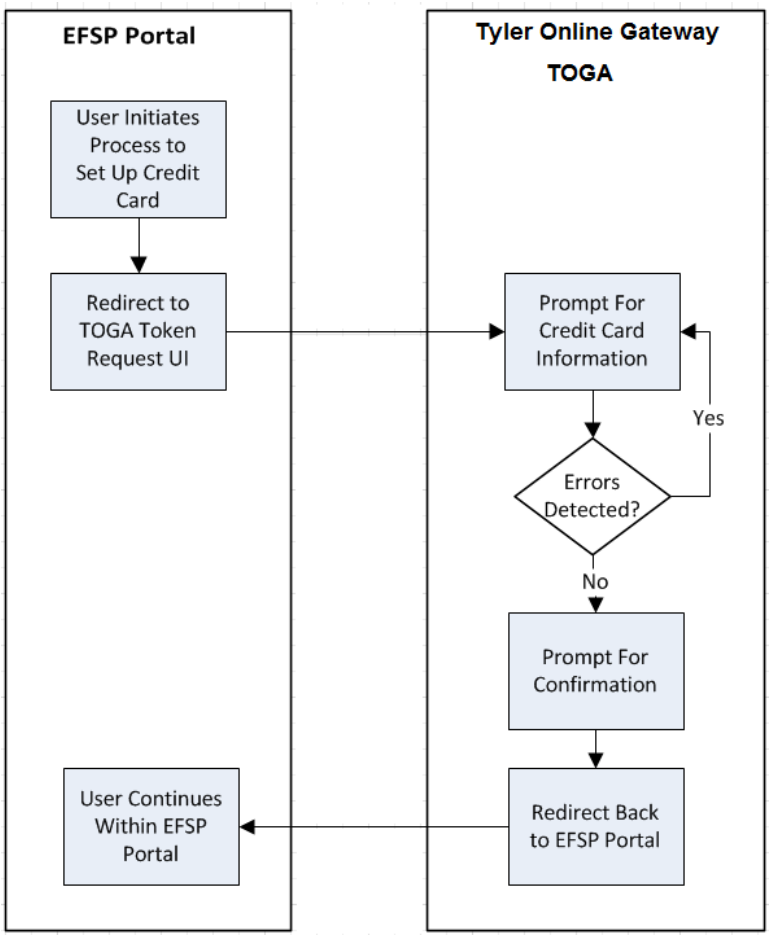

Figure 8 - EFSP Integration with TOGA for a Token Request

- Prompt for credit card information

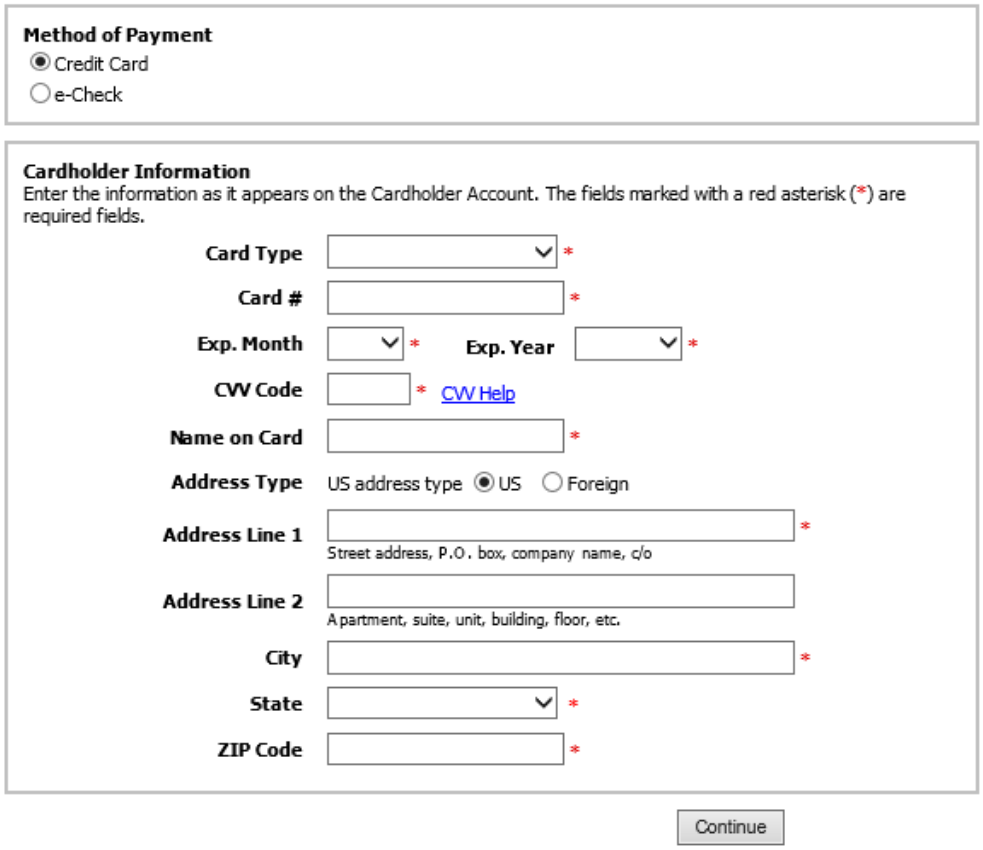

Figure 9 - Prompt for Credit Card Information

- Notice of an error when submitting the form

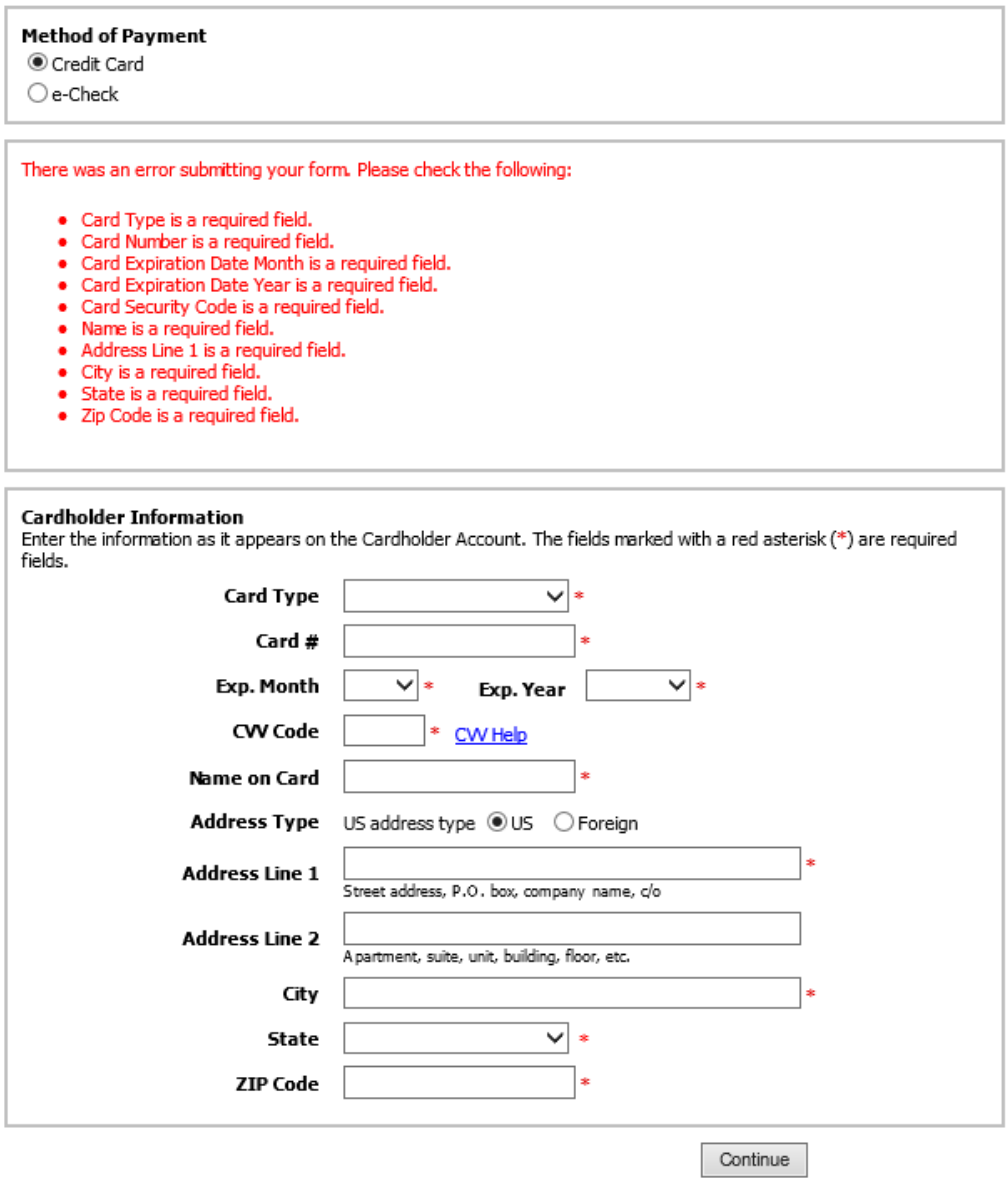

Figure 10 - Notice of Errors Detected

- Confirmation of Card Registration

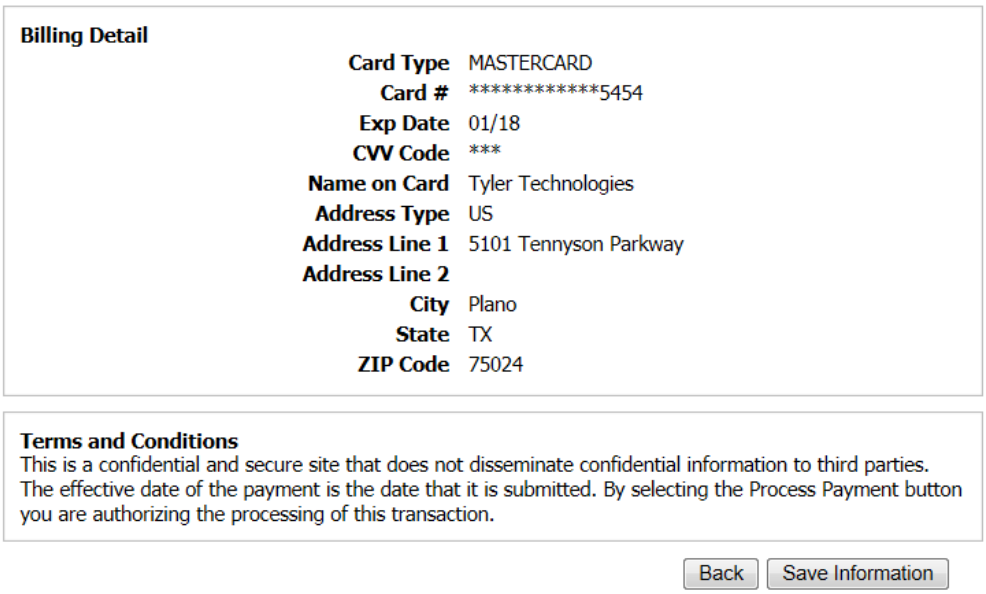

Figure 11 - Confirmation of Card Registration

- Prompt for credit card information

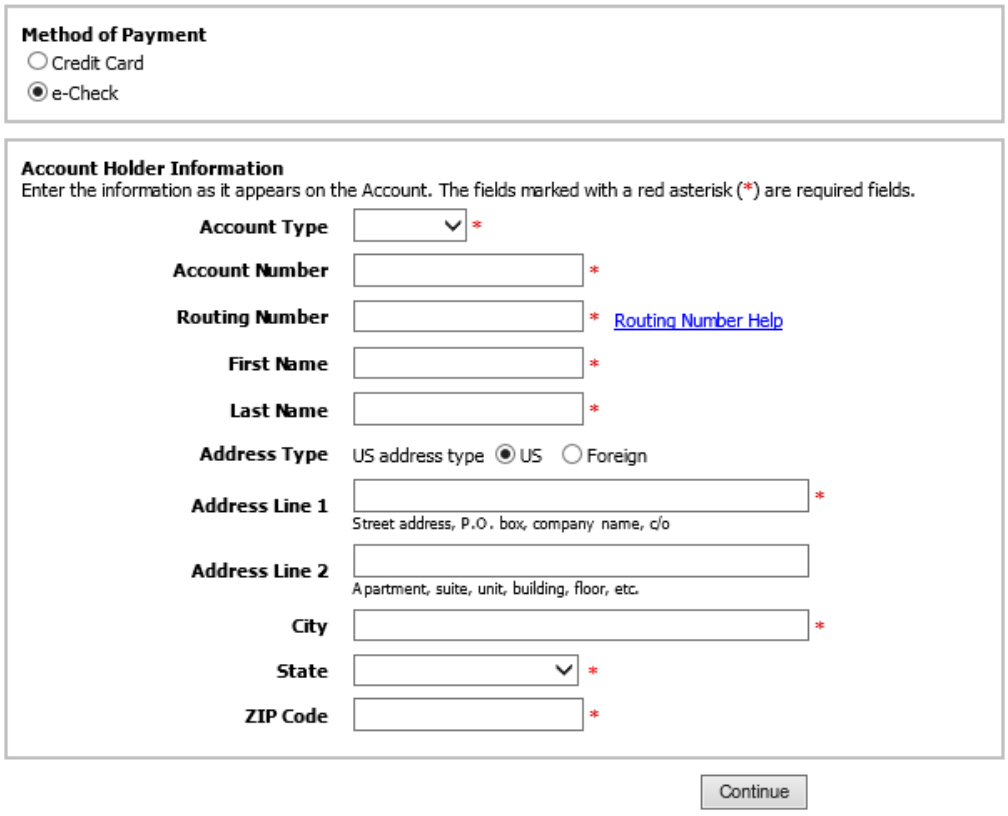

Figure 12 - Prompt for e-Check Information

- Notice of an error when submitting the form

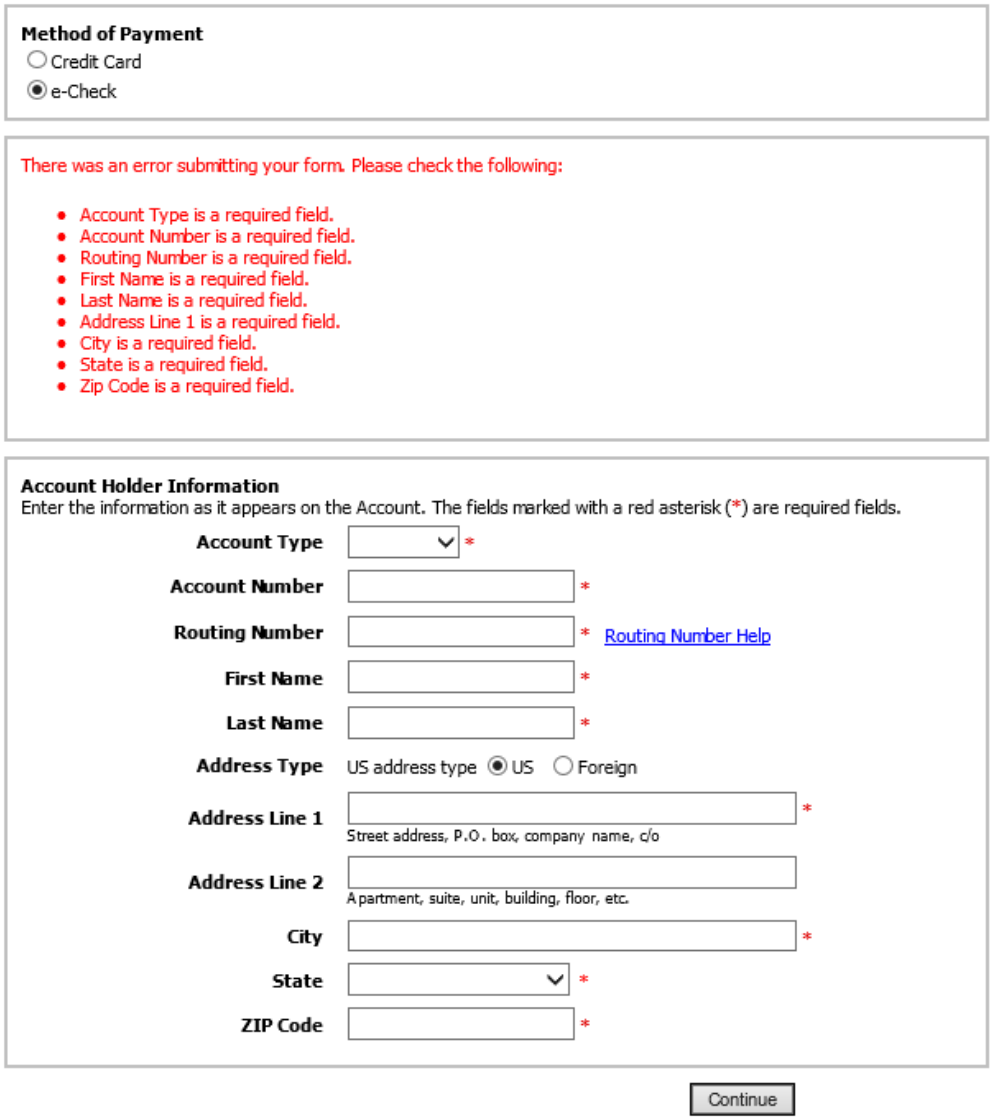

Figure 13 - Notice of Errors Detected

- Confirmation of Card Registration

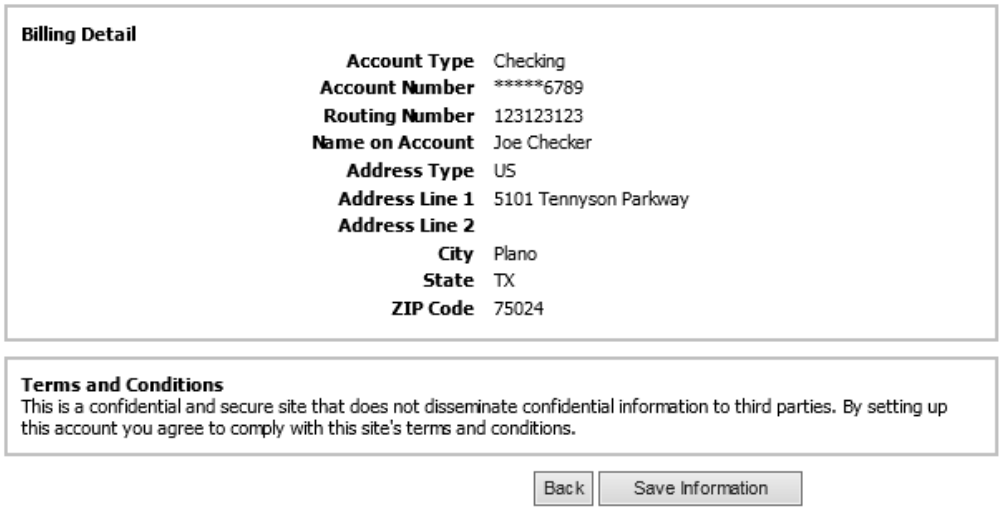

Figure 14 - Confirmation of e-Check Registration

Token Request Processing

The processing of a token request is through an HTTP POST operation to an ASP.NET web page. The specific URL for this web page is provided in conjunction with the rollout of the test and production systems.

TOGA Web Page

The TOGA web page expects XML input through the form element `RequestXML`. To pass the necessary information to TOGA, use the elements defined in this table.

Element Name	Type	Description
PaymentRequest	N/A	Container element for the request.
ClientKey	String	Send the value <<ClientKey>>. The ClientKey is assigned during implementation of a particular jurisdiction.
TransactionID	String	Identifier for this transaction returns in the `ResponseXML` after the payment token is generated. Used to cross reference the token to the request. `TransactionID` is a maximum of 22 characters, with the first eight characters unique. For example: `201001020111111`.
RedirectURL	String	URL used by TOGA to redirect the filer browser to the EFSP portal.

Element Name	Type	Description
Amount	Decimal	Send a value of -1.
GetToken	Integer	Send a value of 1.

Sample Token Request XML

This is *only* a registration of the credit card information; therefore, no amount is charged.

```
<PaymentRequest>
  <ClientKey>ClientKey</ClientKey>
  <TransactionID>201001020111111</TransactionID>
  <RedirectURL>https://www.efsp123.com/return.aspx</RedirectURL>
  <Amount>-1</Amount>
  <GetToken>1</GetToken>
</PaymentRequest>
```

Figure 15 - Token Request XML Sample

Sample Token Response XML

Regardless of whether registering a credit card or an e-Check, the token returns in the `ResponseXML` in the `PayorToken` element.

```
<PaymentResponse>
  <ClientKey>Tyler-Internal-Client-Key</ClientKey>
  <TransactionID>20v101028001</TransactionID>
  <Amount>-1.00</Amount>
  <ReferenceNumber />
  <PayorToken>25233650</PayorToken>
  <PayorName>Guybrush Threepwood</PayorName>
  <TenderDescription>MASTERCARD XXXXXXXXXXXX5454</TenderDescription>
  <ExpirationMonth>3</ExpirationMonth>
  <ExpirationYear>2015</ExpirationYear>
  <PaymentTimestamp>2/19/2013 5:42:10 PM</PaymentTimestamp>
</PaymentResponse>
```

Figure 16 - Token Response XML Sample: Credit Card

The token returns in the `ResponseXML` in the `PayorToken` element.

```
  <PaymentResponse>
    <ClientKey>Tyler-Internal-Client-Key</ClientKey>
    <TransactionID>3ababa1671xyz</TransactionID>
    <Amount>-1.00</Amount>
    <ReferenceNumber/>
    <PayorToken>55119361</PayorToken>
    <PayorName>Joe Checker</PayorName>
    <TenderDescription>CHECKING ACCT XXXXX3123</TenderDescription>
    <ExpirationMonth>0</ExpirationMonth>
    <ExpirationYear>0</ExpirationYear>
    <PaymentTimestamp>8/27/2015 5:00:33 PM</PaymentTimestamp>
  </PaymentResponse>
```

Figure 17 - Token Response XML Sample: e-Check


tyler
technologies