

White Paper: Quik! Forms Security

*An Efficient Technology Inc White Paper
December 2014*



TABLE OF CONTENTS

FOREWORD3

INTRODUCTION3

 ASSUMPTIONS.....3

Definition of Terms3

THE EVOLUTION OF FORMS.....4

 FROM PAPER TO PDF TO HTML4

QUIK! FORMS SECURITY MODEL5

 HOW FORMS ARE GENERATED5

Option 1: Install Quik! On Your System5

Option 2: Use Quik! As A Web Service.....6

Process Flow Explained.....6

 FORM SECURITY IN DETAIL.....7

QUIK! INFRASTRUCTURE SECURITY.....10

SUMMARY OF QUIK! FORMS SECURITY MODEL.....10

ABOUT EFFICIENT TECHNOLOGY INC12

 ABOUT THE AUTHOR: RICHARD WALKER.....12

FOREWORD

This white paper discusses the security architecture and methods used with Quik! Forms throughout the lifecycle of a Quik! Form. While this paper attempts to provide a clear understanding of the Quik! security model, if anything is unclear please reach out to our executive team.

INTRODUCTION

When a form is used what happens to the data on that form? This paper will answer that question in multiple ways in relation to how Quik! Forms addresses the concerns over:

- data ownership
- transporting data
- ensuring data integrity
- data access and persistence

Efficient Technology Inc (ETI), designed Quik! to enable forms automation at all levels, from a simple form to a complex workflow involving forms. Data security is of the utmost importance to ETI and the Quik! line of products. This paper does not intend to be a marketing or sales tool to promote ETI's products but rather to discuss the methods, concepts and options available to secure data on Quik! Forms.

Assumptions

Readers of this white paper should have a basic understanding of forms automation, modern security models and internet-based transportation of data. The technology concepts discussed in this paper do not require a technology background, but will be easier for IT professionals to understand, with the intent of making this topic understandable to all readers.

Definition of Terms

- **Application** means any system, app or other software product that makes use of the Quik! software to deliver forms to end users
- **Customer** means the company who is licensed to use Quik!
- **Forms** means any document output by Quik! (e.g. form, application, contract, agreement, marketing document, etc.)
- **User** or end-user means any person who is using a form output by the customer's application

THE EVOLUTION OF FORMS

From Paper to PDF to HTML

Paper forms have been around for as long as paper has been mass-produced and used in business transactions. A paper form is often viewed as the most cost-effective to produce and easiest to use. As computers became commonplace in businesses the desire to automate paper-based processes became a clear initiative for most companies. While it may appear that we use more paper today than we did decades ago, computer automation has eliminated much of the old paper that was used (do you remember the green accounting general ledgers or carbon paper pads?). What computer automation has really done is helped paperwork evolve to a point where it becomes more necessary and more cost-effective to automate all variations of paper forms.

PDF, the Portable Document Format, was invented by the Adobe corporation in the 1990's as a way to ensure a document could be transported, viewed and printed across all devices. Today PDF is an industry standard adopted by countless software products and users to facilitate sharing documents and forms. While many different technologies were developed to make it possible to fill out forms on a computer, the PDF forms tools had the largest impact on the computing world and is the de-facto standard for sharing forms.

HTML, the language of web browsers, changed everything. With the advent of the internet, consumers of all types benefited from filling out forms in web browsers. Where PDF took the paper version and made it work online, HTML took any form that no longer had to look like paper and eliminated the paper altogether.

The challenge that Quik! solves, and it's reason for existing, is to make it possible to automate forms in ways that neither paper, PDF nor HTML can accomplish. Quik! takes existing forms (e.g. a PDF form) and makes it interactive, fillable, secure and signable within any web browser by converting the form to HTML, all while retaining the exact look and feel of the original form. As a service, Quik! provides the infrastructure, technology, methodology and service personnel to make forms automation accessible to virtually any user. In essence, Quik! helps forms evolve from paper, PDF and HTML to do more with less effort and cost.

QUIK! FORMS SECURITY MODEL

First and foremost, Quik! is **NOT** in the business of managing, storing, archiving, interpreting or accessing client data on forms. Quik!'s function is to help customers manager their library of forms and to deliver forms to users when they need them and on the device they're using. With this goal in mind, Quik! Forms are designed to keep data in the control of the customer and user.

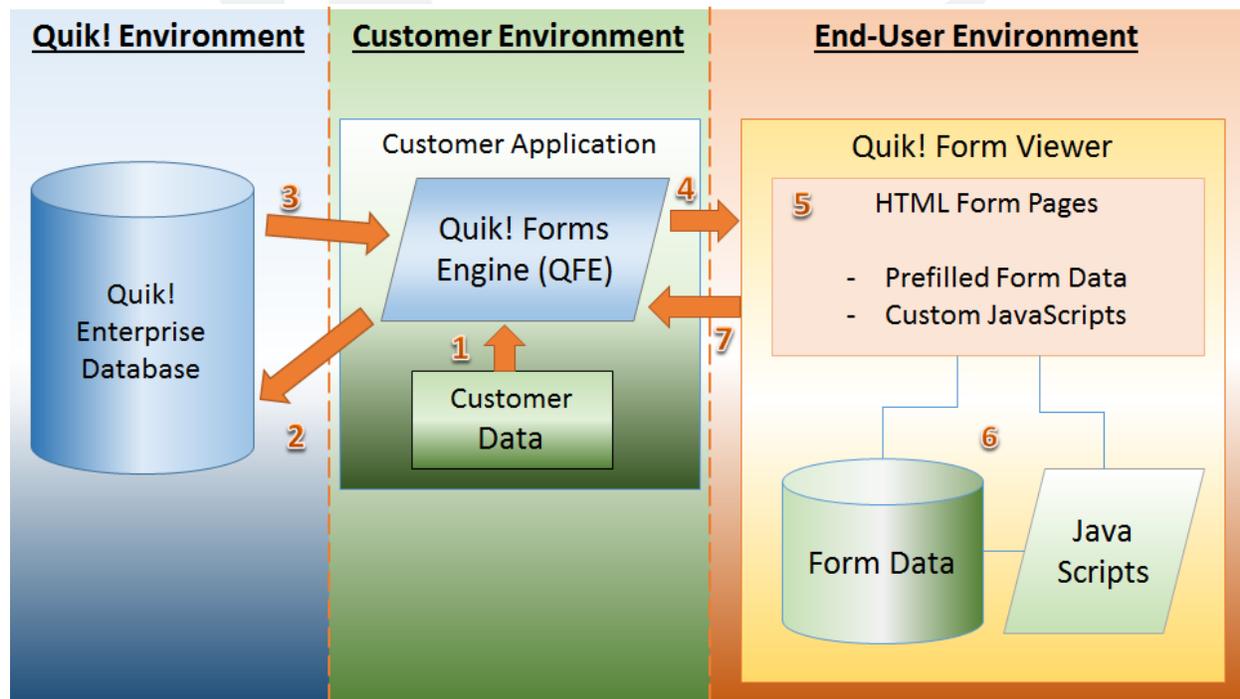
How Forms Are Generated

In order to view a Quik! Form it must first be generated by the Quik! software. By always running a Quik! software component to generate the form the user will always receive the latest form version and never turn in a form that is no longer in service. The Quik! software, at its core, is like the engine of a car: it takes in fuel and creates power, or in Quik!'s terms, it takes in data and outputs a form for the user to see.

Like an engine can be used to power a car, boat, plane, truck and more, the Quik! Forms Engine is used to power websites, mobile apps, desktop software, server-side software and more. Regardless of what application is using the Quik! Forms Engine, the underlying operations are always the same, as illustrated in the pictures below.

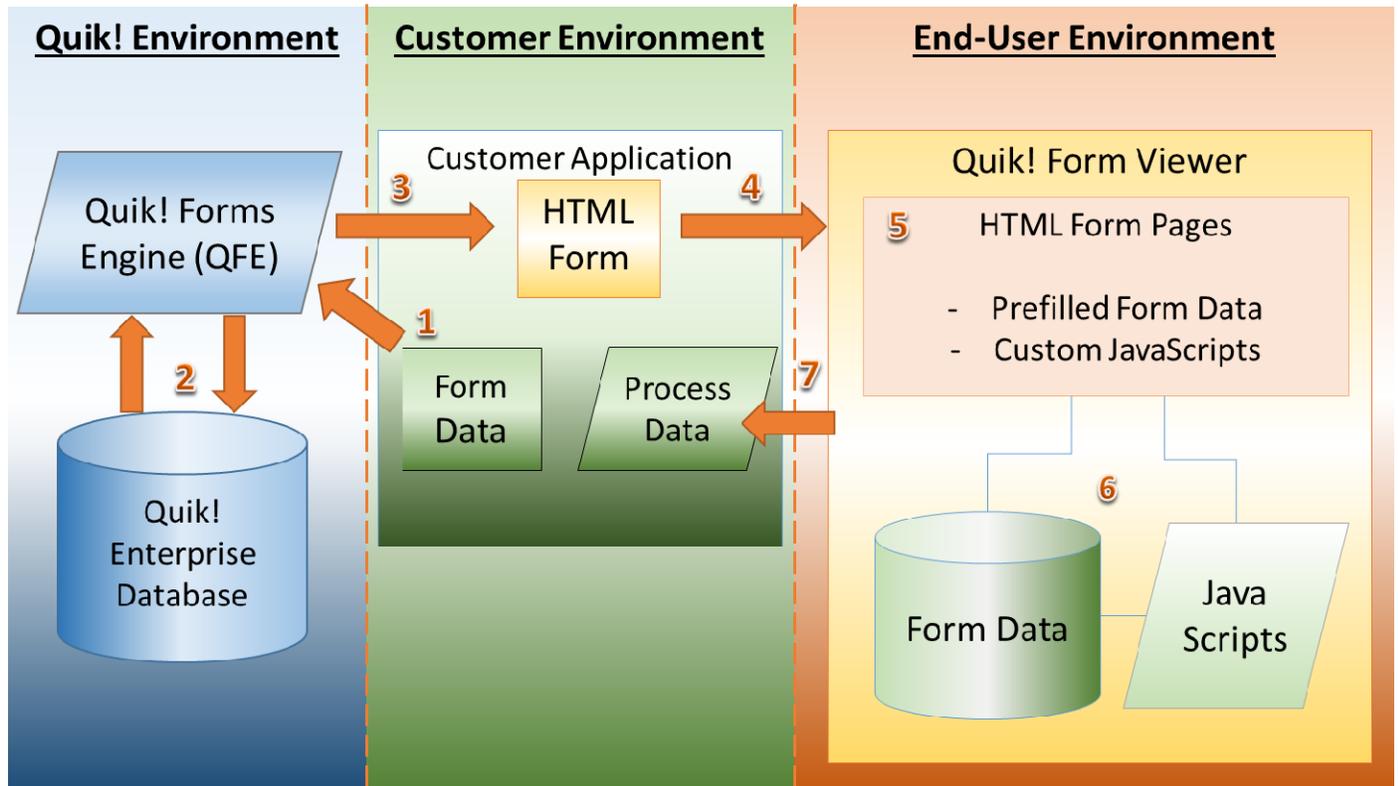
Option 1: Install Quik! On Your System

For convenience, security and control, customers can implement the Quik! software (QuikFormsEngine.dll) directly within their application.



Option 2: Use Quik! As A Web Service

For customers who cannot implement the Quik! software directly within their application, a web service can be used instead. The Quik! Forms Engine web service is the same QuikFormsEngine.dll hosted by Quik! and works essentially the same way as the software.



Process Flow Explained

1. The Quik! software is provided with data to prefill onto forms (DLL or Web Service) and run.
2. The Quik! software calls the Quik! Environment to retrieve the requested forms
3. The forms are sent back to the customer application, populated with the customer data
4. The customer application serves the resulting Quik! form in HTML format to user's browser
5. The forms display prefilled with data
6. The user enters data and completes the form
7. The form can be sent back to the customer application for further processing, signature and storage.

Form Security In Detail

The Quik! architecture ensures the customer's application and the end-user are in control of the data, not Quik!. Quik! provides the underlying forms architecture for applications via a single interface (the Quik! Forms Engine).

With this architecture in mind, the data security for forms is easier to understand and identify.

In steps 1 through 3 of the architecture diagram, the customer application is the start and end point of a round-trip to the Quik! server. The transaction starts at the customer application to ensure the customer application controls the data provided to the Quik! Forms Engine to populate on the form. There are two methods in which the Quik! Forms Engine can be run and both methods are discussed from a data security standpoint:

- Installed Software (QuikFormsEngine.dll - .NET Framework)
- Web Service

Installed Software

When installed inside an application, the data provided to the Quik! software will go straight onto the resulting Quik! Form (step 3). No data is sent to the Quik! server (step 2), only the customer's credentials and the form request is sent so Quik! can validate the customer's rights to generate the requested forms.

Web Service

The Quik! Forms Engine web service method is slightly different than the diagram above in that the actual Quik! Forms Engine software (i.e. the QuikFormsEngine.dll) is hosted on a Quik! server instead of in the Customer Application. This method is optional and often chosen by customers who need a cloud-based solution.

When using Quik! via the web service method, the data provided to the web service is sent to the Quik! web server with the following security protocols:

- 256-bit SSL encryption (standard encryption used for internet traffic)
- Data received on the Quik! server is loaded into the Quik! Forms Engine software residing on the Quik! server (just like in Step 1) and used to create a form (step 3) that is returned directly to the client application.
- The Quik! server:

- Does NOT store, write to disk or archive the form or related data in any way – the generated form that includes any data to prefill onto the form is held in random access memory (RAM) for the duration of the request only and is immediately released after the form is returned to the calling application.
- Does NOT interpret, access or read the data for any logical operations – the data is simply placed into the form as requested.
- Does NOT enable any person, including potential hackers, to view, access or retrieve data on forms in any manner through the Quik! server.

In step 4 of the diagram, the customer application delivers the form output (HTML) directly to the end-user's browser. Since the customer application is in control of this step, the customer is responsible for the security method. For example, the customer will most likely:

- Deliver the forms from within the customer's domain (e.g. <https://www.CustomerURL.com>)
- Deliver the forms via SSL security (i.e. "https://")
- Authenticate the user before a user can generate a form and view it (e.g. some kind of login or user authentication)
- Control whether or not the form is written to the customer's own computers (e.g. web server, local drive, network, etc.)

In step 5 and 6, the end-user is viewing the form in their web browser. By virtue of how web browsers work, the entire HTML output from the Quik! Forms Engine is downloaded by the web browser to the end-user's machine in order to be displayed. This ultimately means that the user's device has the HTML, inclusive of any prefilled data, locally resident on that device's drive. How the user configures their web browser and secures their device determines how secure their data is.

Note: Data on a form is not at the same level of risk as an entire database or organized records. Even if a user's device is compromised, the form data is not managed or organized in a way that is meaningful or valuable to a hacker, nor is there more than a few records at any given time.

Finally, in step 7 a user can choose to print, submit and/or sign the form depending on the customer's configuration of the Quik! form. These three options create different avenues for data.

Print

Printing a form will send the form data to the Quik! server for the sole purpose of creating a PDF that is returned to the user. When the print routine is run, the data is sent securely to Quik! and immediately turned into a PDF that is returned as a response.

The form data sent to the print service is sent with the following security protocols:

- 256-bit SSL encryption (standard encryption used for internet traffic)
- Data received on the Quik! server is converted into a PDF form that is returned directly to the client application.
- The Quik! server:
 - o Does NOT store, write to disk or archive the PDF or related data in any way – the PDF that includes any form data is held in random access memory (RAM) for the duration of the request only and is immediately released after the PDF is returned to the user's browser.
 - o Does NOT interpret, access or read the data for any logical operations – the data is simply placed into the form as requested.
 - o Does NOT enable any person, including potential hackers, to view, access or retrieve data on forms in any manner through the Quik! server.

Submit

If the submit button is enabled on a form, the customer must provide a URL to submit the data to. That way when the submit button is clicked by a user the form will send the data directly to that URL. Since this URL is not hosted or controlled by Quik! in any way, the security of this URL is determined by the customer. Quik! recommends using SSL with 256-bit encryption on a secure server.

Sign

Depending on how customer chooses to implement e-signing, clicking the Sign button may send the form data to the Quik! server for the sole purpose of creating a PDF that is used with the e-signing service. When the sign routine is run, the data is sent securely to Quik! and immediately turned into a PDF that is used in the signature process.

The form data sent to the sign service is sent with the following security protocols:

- 256-bit SSL encryption (standard encryption used for internet traffic)

- Data received on the Quik! server is converted into a PDF form that is returned directly to the client application or loaded directly into the e-sign system.
- The Quik! server:
 - o Does NOT store, write to disk or archive the PDF or related data in any way – the PDF that includes any form data is held in random access memory (RAM) for the duration of the request only and is immediately released after the PDF is returned to the user's browser.
 - o Does NOT interpret, access or read the data for any logical operations – the data is simply placed into the form as requested.
 - o Does NOT enable any person, including potential hackers, to view, access or retrieve data on forms in any manner through the Quik! server.

The sign button does require the service of a third-party e-sign vendor. Each e-sign vendor is responsible for their security protocols and Quik! recommends doing diligence on each vendor before choosing one. The e-sign partner(s) Quik! works with are well-established companies with a proven track-record of strong security.

QUIK! INFRASTRUCTURE SECURITY

As a policy, details about the Quik! operating infrastructure are not made readily public in order to ensure malicious persons are not informed for how to attack Quik!'s infrastructure. What is publicly known about Quik!'s infrastructure is as follows:

- Quik!'s hosted environment is provided by Amazon Web Service
- Amazon is a world-class cloud hosting provider trusted by the US Government, top financial institutions and companies world-wide.
- For more details, please visit: <http://aws.amazon.com/>

In addition, Quik! follows industry standards and best practices to maintain security and service uptime. Quik! goes through regular, periodic reviews by both internal and external parties to ensure a strong and secure operational environment.

SUMMARY OF QUIK! FORMS SECURITY MODEL

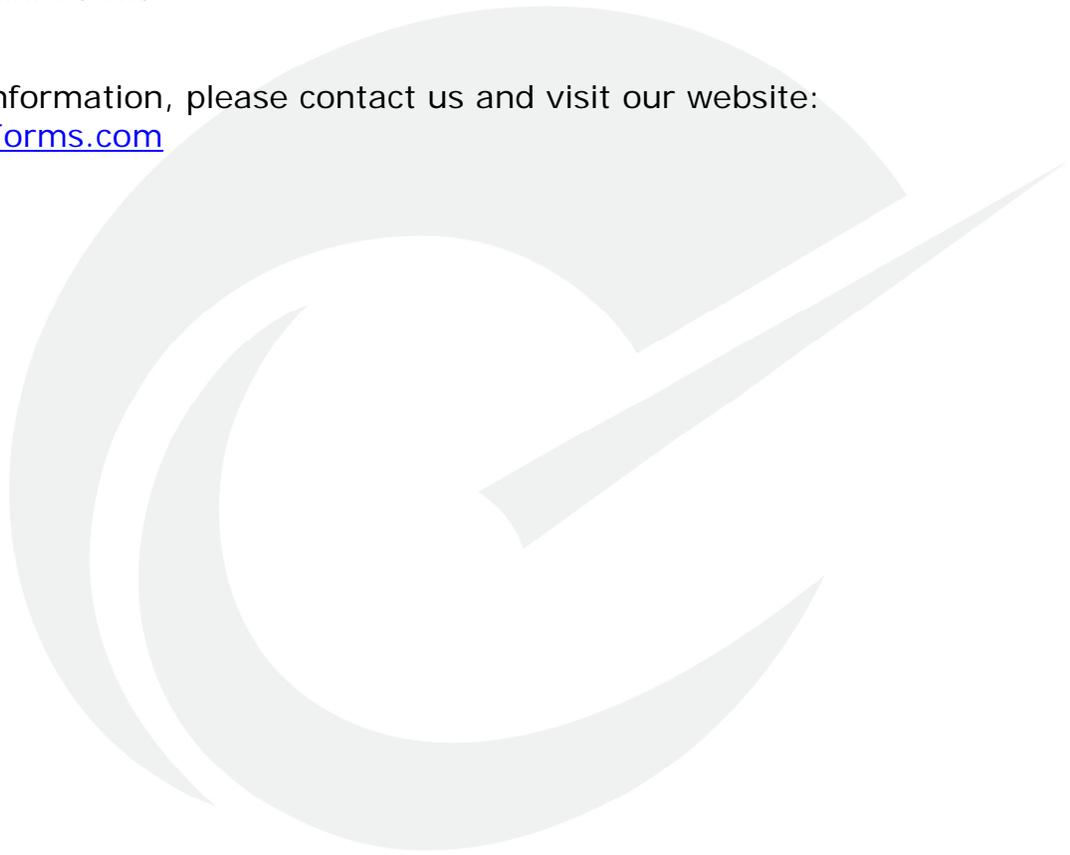
Forms generated by Quik! are designed to:

- Give customers ownership over their data
- Give customers control over their user experience

- Use industry-standard or best-in-class security protocols when transporting data
- Ensure the integrity of data throughout the life-cycle of a form
- Eliminate the need for Quik! to store, archive, access, interpret, read or persist a customer's data in any way.

The bottom line at Quik! is we are people like you who care about our own data security and want to sleep at night knowing we are safe. We sleep well and will do everything possible to ensure you do too when it comes to trusting Quik! Forms.

For more information, please contact us and visit our website:
www.QuikForms.com



ABOUT EFFICIENT TECHNOLOGY INC

Efficient Technology, Inc. (ETI) provides enterprise forms-based workflow automation solutions that accommodate dynamically changing business rules, forms and workflow. ETI is the developer of Quik!, the industry-leading forms management and forms-enablement solution. By improving business processes, increasing efficiency and saving time Efficient Technology Inc makes work flow. With over 60,000 end-users ETI's customers range from Fortune 500 companies to small businesses. Efficient Technology Inc is proud to be a carbon-negative company saving over 1,000 trees each year. Visit www.EfficientTech.com

About The Author: Richard Walker



Richard Walker is the Efficient CEO and co-founder of Efficient Technology Inc where he leads customers to enterprise efficiency. Prior to starting ETI, Richard was a senior consultant with Arthur Andersen's Business Consulting unit, implementing large enterprise technology solutions for Fortune 500 companies. He also spent three years as a registered securities representative and has over 10 years of experience in financial services working in various capacities at: ING, Transamerica, PaineWebber, John Hancock and Donaldson, Lufkin and Jenrette. Richard earned his B.S. degree in Business Administration – Finance from University of Southern California, and is the author of many articles and the www.EfficientCEO.com blog.



Making Work Flow

White Paper: Quik! Forms Security

December 2014

Author: Richard Walker

Efficient Technology Inc

811 N Catalina Ave, Suite 3104

Redondo Beach, CA 90277

USA

877-456-QUIK (7845)

www.EfficientTech.com

Copyright © 2014 Efficient Technology Inc. All Rights Reserved

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Quik! is a registered trademark of Efficient Technology Inc. Other names may be trademarks of their respective owners.

 *Efficient Technology has helped customers save over 10,000 trees. Join us in saving more.*