qClient™

qClient SDK | KDP
Key Management-Focused Software Development Kit and KMIP Development Platform

Overview
qClient is a highly capable and vendor neutral client software development kit which provides an ideal solution for developers wanting to integrate powerful cryptographic key and random number management.

Integrating Key Management
qClient enables developers to integrate QuintessenceLabs’ key and policy manager and true random number generator to deliver the strongest foundation for the most demanding encryption needs. It comes with all the components required for fast and efficient integration including libraries, header files and sample source code.

The OASIS Key Management Interoperability Protocol (KMIP) is a communication protocol that allows the manipulation of keys on a key management server. It facilitates the deployment of secure encryption across an organization by allowing cryptographic key management and random number management to be quickly and easily integrated into any application.

qClient supports a general purpose key management API for full KMIP functionality. Developers don’t need to be KMIP experts to take advantage of qClient – the main API has a concise number of functions for acquiring, creating, checking and otherwise maintaining managed keys, certificates and related data.

qClient Deployment
qClient SDK: qClient can be delivered as a standalone interoperable client to allow developers to integrate key and random number management.

qClient KMIP Development Platform (KDP): This combines qClient SDK with a not-for-resale license for the qCrypt 200V virtual machine.

qClient is also a component product of the QuintessenceLabs data production platform, included in all qStream or qCrypt deployments.

qClient Capabilities
Proven compatibility with KMIP servers lets qClient take the guesswork out of interoperability, while a powerful key management API lets developers work without losing functionality. qClient comes with all the components required for fast and efficient integration, including binary libraries, header files, additional APIs for key management and PKCS#11, plus source code samples and detailed documentation.

qClient is designed to be embedded in applications, cross-platform interfaces, and platform-specific implementations. It can operate with secure transport, synchronous and asynchronous operations, and file system and network I/O.
### SPECIFICATIONS

**qClient™**

**qClient SDK | qClient KDP**

<table>
<thead>
<tr>
<th>Key Features</th>
<th>qClient SDK</th>
<th>qClient KDP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allows for rapid development of key management applications</td>
<td></td>
<td>Includes not-for-resale qCrypt 200V virtual machine license*</td>
</tr>
<tr>
<td>Supports OASIS KMIP and PKCS#11 protocols</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Interoperates with qCrypt and compliant third-party key management servers</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Facilitates rapid development, relieving developers of the need for deep KMIP or PKCS#11 knowledge</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### KMIP & PKCS#11

- OASIS KMIP: Conformant with standards 1.0/1.1/1.2/1.3/1.4/2.0
- Supports all KMIP profiles and transport protocols: TTLV, TTLV over HTTPS, JSON, XML
- PKCS#11 version 2.40, Baseline Provider and Extended Provider profiles, Authentication Tokens

#### Languages

- qClient is implemented in C
- .NET and Java APIs
- Other APIs available upon request

#### Supported Platforms

- 32 bit Windows 7, 8+, Server 2008 R2+
- 64 bit Windows 7, 8+, Server 2008 R2+
- 32 bit Linux x86 Kernel 2.6+
- 64 bit Linux x86_64 Kernel 2.6+
- ARMv8 Debian 4.9130
- 64 bit ESXi 6.5, 6.7

#### Supported KMIP Managed Objects

- Certificate
- Opaque Object
- Private Key
- Public Key
- Random Object
- Secret Data
- Symmetric Key
- Template

#### Supported KMIP Operations

- Objects Establishment
- Object and Attribute Retrieval
- Object and Operation Accessibility
- Object Lifecycle
- Attribute Management
- Object Renewal
- Asynchronous Operations
- Server Initiated
- Discovery
- Crypto Operations
- Split Key Operations

#### Supported KMIP Profiles

- Baseline
- Symmetric Key Lifecycle
- Basic Symmetric Key Foundry
- Intermediate Symmetric Key Foundry
- Advanced Symmetric Key Foundry
- Asymmetric Key Lifecycle
- Basic Cryptographic
- Advanced Cryptographic
- RNG Cryptographic
- Objects Establishment
- Object and Attribute Retrieval
- Object and Operation Accessibility
- Object Lifecycle
- Attribute Management
- Object Renewal
- Asynchronous Operations
- Server Initiated
- Discovery
- Crypto Operations
- Split Key Operations

#### Supported PKCS#11 Profiles

- Baseline Provider
- Extended Provider
- Authentication Token ("KMIP", "Keystore software" provided with support for customers to provide their own token)
- Advanced Symmetric Key Foundry
- Asymmetric Key Lifecycle
- Basic Cryptographic
- Advanced Cryptographic
- RNG Cryptographic
- Objects Establishment
- Object and Attribute Retrieval
- Object and Operation Accessibility
- Object Lifecycle
- Attribute Management
- Object Renewal
- Asynchronous Operations
- Server Initiated
- Discovery
- Crypto Operations
- Split Key Operations

* Refer to qCrypt 200V data sheet for more information

---

©2019 QuintessenceLabs. All rights reserved.