KeyWatcher System Integration Methods

There are currently 2 KeyWatcher products available:

- KeyWatcher Illuminated
- KeyWatcher Touch

Please visit our website: www.morsewatchmans.com for more information on the systems and their differences

- **KW Illuminated System**:

  There are two methods by which integration can be done with the KW Illuminated/KW III system.

  - KeyWatcherXML
  - Hardware communications protocol

  - KeyWatcherXML:
    - This package includes a small server app that takes care of any hardware communications with the KeyWatcher. It is not able to transactions collection however.
      - Most database commands are available that would normally be available in KeyPro III.
      - Live Key reports are available (Key In Use/Key Inventory/Key Status)

    - This app works by monitoring an input folder where XML files are saved by the 3rd party application. The files contain the proper XML layout and commands for the process that is being done, such as adding users, changing the names of keys, or running a live key report. Once the file is processed, the KeyWatcherXML app will communicate with the KW system, and then save an output folder for the 3rd party app to process and parse for information showing the operation was completed successfully or other information that was requested.

    - This is a much lower cost option compared to the Comm. Protocol, and should require less time.

  - KWIII Comm. Protocol
    - This is the hardware communications protocol used by our KeyPro III software to communicate with the KWIII CPU. Utilizing this method of integration with the KeyWatcher will enable software developers to create a piece of software capable of all the same function as available in KeyPro

    - It is however much more difficult to develop the software since all the communications with the system will fall on your programmers.
      - This method would allow 3rd party software to collect transaction history and save them in its own DB.

The downside to either of these methods is that the KeyPro III software must be used in a more specific way to avoid syncing old information down to the KeyWatcher, since the KPIII Database would not be updated.

- **KeyWatcher Touch System:**

  The KWTouch system is much different in that it utilizes a server application that is the “heart” of the system. The KWTServer application is responsible for all aspects of the system, including managing the DB and synchronizing the KWTouch systems. In this system, all transactions are pushed up to the server, rather than retrieved. This is our newest system and is the future of the KeyWatcher product.

  Integration in this product has been designed into the KWTServer application via an API.

  - SDK / API Integration:
    - This integration is much more straightforward and should prove to be an easier, faster integration to design and get functional. This integration will also be much more capable. The KWTServer app has a 3rd party API interface that allows for all the same functionality that is available in TrueTouch (Our client application). All reporting, transaction inquiries, etc. can be done through that interface – and there is no hardware level communication needed.
    - SDK Document: This document provides all information needed for working with the KWTServer API, and also includes some sample code. This document will continually be updated as functionality to the system is added.

  - Database and system integrity:
    - The KWTouch system database will always be the DB that is altered by any 3rd party client software logging in. This allows all clients to always be up to date and working with all the latest programming and transaction.
    - System Integrity - Since all changes are processed through the KWTServer, we are able to prevent bad or incorrect/invalid information from being saved to the system with error feedback.

Recommendations:

For either integration, we strongly recommend that a demo system be purchased for development.

- For the KWIII system, using a protocol analyzer will be necessary if working with the comm. protocol.

Support:

Engineering level support is available for an extended cost; More information is available upon request.

Non-Disclosure Document

Prior to any software being distributed (KWXML) or documentation being shared (Comm. Protocol/KWT SDK), all parties involved will be required to sign a non-disclosure document and provide us with a description of the work being done/purpose.