|  |
| --- |
| YAle School of Architecture |
| Make the Cut!™ PaperCut™ Plugin |
| PaperCut™ Tracking Plugin Installation and Setup Guide |
|  |
| **McMorran, Patrick** |
| **1/29/2013** |

|  |
| --- |
| This document contains explanation of code and how to setup the plugin to track a KNK Max Air cutter as though it was a printer in PaperCut™. |

Table of Contents

[Introduction 2](#_Toc348519669)

[License 2](#_Toc348519670)

[Functional overview 2](#_Toc348519671)

[Prerequisites 3](#_Toc348519672)

[Virtual Queue Configuration 3](#_Toc348519673)

[Windows Server 2008 R2™ 3](#_Toc348519674)

[PaperCut Application Server Virtual Queue Configuration 5](#_Toc348519675)

[Configuration of the Machine attached to the KNK MAXX AIR 6](#_Toc348519676)

[The Authentication Plugin 6](#_Toc348519677)

# Introduction

This plugin was written in C# with the intention of tracking the KNK MAXX AIR cutter, which requires use of a specialized piece of software in order to cut with it. This piece of hardware does not make use of the print spooler, and instead is directly controlled by a computer over a COM interface using either 9-pin Serial, USB, or Bluetooth connection. PaperCut™ has a print monitoring service (the print server roll) which tracks print jobs by analyzing spool files, and thus is not designed to monitor COM activity. The developer of Make the Cut!™ has added the ability to use custom authentication plugins with the software, allowing for the creation of this plugin to ensure machine usage could be tracked using PaperCut™. This software and accompanying documentation is public domain, and is accompanied by a license explaining such. PaperCut™ and Make the Cut!™ trademarks are property of their respective owners, and the software’s are subject to separate licensing terms.

## License

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>

## Prerequisites

To install and setup this plugin requires the following is required:

* Access to the PaperCut™ application server API Key
* A virtual print queue
* Microsoft™ .net Framework version 2.0 or Higher on the Computer attached to the KNK MAXX AIR
* Administrative Access to the Registry on the Computer attached to the KNK MAXX AIR

## Functional overview

Make the cut has implemented a few registry keys that when populated with data it will run a program to check for authentication when the cut button for the KNK MAX AIR is hit. If the application exits with a return code of 0 then Make the Cut! ™ presumes that the user has been authenticated and starts the cutting portion of the software. This works similar to a printer, where the print server verifies that the user can be authenticated is permitted to print. Should the authentication plugin run and exit with a non-zero error code, then the user is not permitted. Make the Cut!™ will then pop up an error message that contains whatever text the authentication plugin sent to the Console or Standard out to inform the user why they are not permitted to use the machine, such as insufficient funds or a locked account.

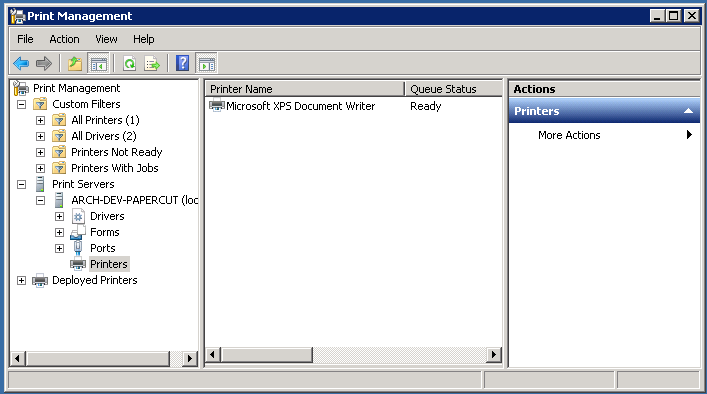
This plugin makes use if the XML Web Services API provided by PaperCut™ to authenticate cuts over the network. Written in C#, the authentication plugin pulls the information about needed to record the “Print Job” such as logged in user, computer name, etc and then makes an XML Remote Procedure Call to the defined PaperCut™ application server. Using the “api.processJob” command, cuts sent on the machine then show up in the user’s print log, attributed to predefined print queue in PaperCut™ that is not attached to an actual machine, i.e. a virtual queue. Should the job fail to be processed, the authentication plugin returns a non-zero exit code and prints an error message to the console to inform the user why the job was denied.

# Virtual Queue Configuration

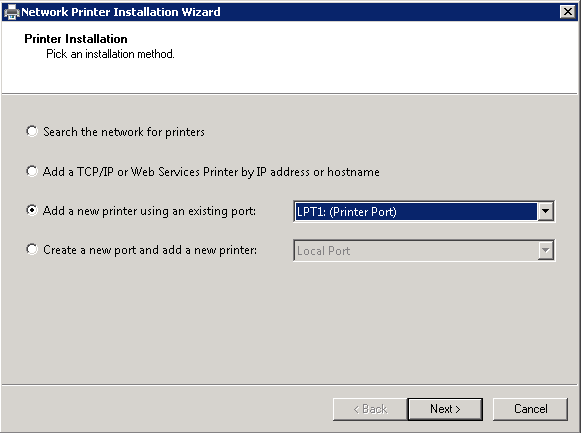
This plugin requires that a Virtual Queue be created for jobs to be submitted to, which will require access to both a print server and the PaperCut™ Application Server. Instructions are available below on how to create a Virtual Print Queue for the authentication application to use.

## Windows Server 2008 R2™

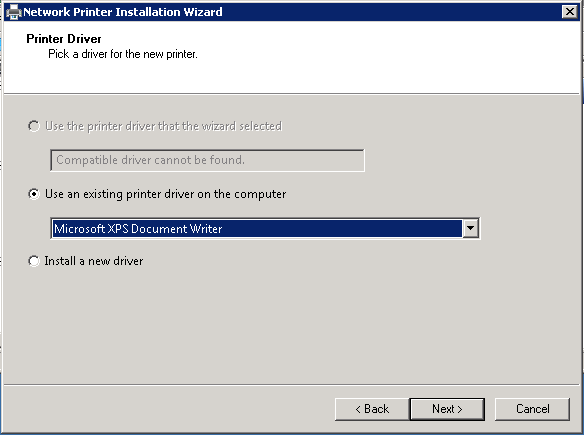
Open the Print Management Console and navigate to the printers section.



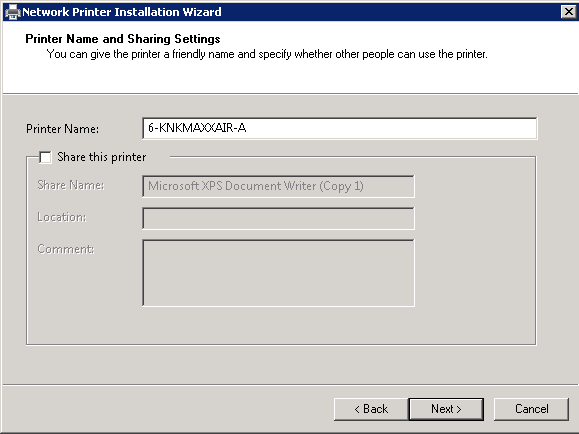
Select More Actions and hit Add Printer…



Add a printer using an existing port, using a port that is not actually in use by any other printer.



Use and existing driver to save on time, as the selection is arbitrary.



Finally enter a name for the print queue following whatever convention has been decided on for your equipment. Uncheck share this printer, as this queue will not be used by any actual end users. Click next to see the confirmation page and verify that all details are correct. Then click next button and the finish button to complete the installation.

## PaperCut Application Server Virtual Queue Configuration

The new queue that was created on the print server should now have popped up onto the application server. If it has not, send a test page to the queue on the print server to trigger the print monitor software to create the queue. Once PaperCut™ has created the queue delete the one on the actual print server, leaving the queue in PaperCut™ to serve as the printer users will be tracked with. Next set the price on the queue. The plugin was designed to implement simple printing, charging a flat rate per cut, and does not return information about the job that can be used for costing each individual job. This plugin could always be extended to do just that.

Next, generate a complex API Key for the web services. The password for the built in administrator account can be used instead, however it is better practice to use an API key to utilize the system, as using the administrator password over an XML-RPC transmission could lead to a compromise of the system. Once you have a secure key (Generating an SHA1 Hash is fine) go to Options -> Config editor (advanced). Use the quick find to locate the “auth.webservices.auth-token”, updating its value with the newly generated key. Last go to Options -> General -> Security and set the Allowed XML Web Services callers to include the IP address of the client machine that will be reporting jobs. This prevents commands being issues from unauthorized computers even in the event that they discover a valid API key or administrator password.

# Configuration of the Machine attached to the KNK MAXX AIR

Now the client machine need be configured in order to authenticate with the PaperCut™ server. This first requires that you configure and compile the authentication module that will be used by Make the Cut!™ software. Ensure that the server has been fully configured so that tests can be done along the way during the client configuration.

## The Authentication Plugin

Install the Microsoft™ .net Framework 2.0 or later, which will be needed in order to compile the c# code. There are then two resources that must be obtained, an accompanying c# file called *ServerCommandProxy.cs* located in the */server/examples/webservices/* directory inside the PaperCut™ application server program files directory. Copy the *ServerCommandProxy.cs* into the same working directory that has the *MaxxairAuth.cs* file which accompanies this ReadMe. Then go to <http://xml-rpc.net/> and download the latest production release. In the zip folder there should be a *CookComputing.XmlRpcV2.dll* file that will be used by the *ServerCommandProxy.cs* to communicate with the PaperCut™ server; copy this file into your working directory.

Open the *MaxxairAuth.cs*, navigate to the user configuration area and fill in the variables with your values. *authToken* can either be the built in administrator account password, or the web services API key that was set earlier in this tutorial; it is suggested that you use the API key for security purposes.