

HTML to Image conversion

Posted At : April 27, 2012 2:48 PM | Posted By : Tim Garver

Related Categories: How Too's

This is something I have been struggling with for a few months. I have tried the suggestions of others and nothing really worked the way I wanted it to. The results were either a failure all together or an image that was either un-readable or so small that it was un-usable.

So I found the other day a few very good command line utilities that work very well. Up until now I had only been attempting native ColdFusion code or Java code called from ColdFusion. I had wanted to stay away from a command line tool simply because you have to have permission from your host to run the cfexecute tag. And since I was on a shared host, I knew this approach would not work at all.

Then I had a brain fart. what if I spun up a [AWS EC2](#) instance and ran some CF code to call cfexecute? It turns out this is very simple to do and with the Free versions, its cheap too.

So how did I do all this? First you will need an AWS account, the link above will help with that. Once you have your account with AWS, simply launch an EC2 instance, you can use the quick launch make sure you choose an image with a web server either IIS or apache. It takes a few minutes to spin all the way up. Once its up, get your password and log into the box so you can install ColdFusion.

After you have your box up and running with CF installed, its just a matter of setting up your conversion page.

Next [Download CutyCapt](#) on the server and put it were ever you like.

Then under your web root, setup your conversion cfm file. We will be posting a few form vars to it, then it will execute our converter and save the image.

```
<cftry>
<cfparam name="form.key" default="">

<cfif form.key EQ "abc123" and isDefined("form.url") AND isDefined("form.imgname")>
    <cfset u = " --url=#form.url#">
    <cfset o = " --out=C:\inetpub\wwwroot\img\#form.imgname#">
    <cfexecute name="C:\CutyCapt\cutycapt.exe" arguments="#u# #o#" />
true
<cfelse>
false
</cfif>
<cfcatch type="any">
false
</cfcatch>
</cftry>
```

Its pretty straightforward. for security i pass it a key and verify it is correct, i know its not very strong but you get the point. then if the other form vars are there we

continue. we setup our arguments from CutyCapt and then execute the exe file.

This will create our image in the directory we specify. Ok that part is done and now back to your web server.

I setup a CFC for this with my other image utilities.

```
<cffunction name="getURLScreenShot" access="public" output="false" returntype="boolean">
  <cfargument name="url" required="true" />
  <cfargument name="imgname" required="true" />
    <cfargument name="userid" required="true"/>

  <cfhttp url="http://ec2url/convert.cfm" method="post" >
    <cfhttpparam name="key" value="abc123" type="formfield">
    <cfhttpparam name="url" value="#arguments.url#" type="formfield">
    <cfhttpparam name="imgname" value="#imgname#" type="formfield">
  </cfhttp>
  <cfset sleep(5000)>
  <cfif cfhttp.FileContent EQ "true">
    <cfif NOT directoryExists("\wwwroot\images\userimages\#arguments.userid#")>
      <cfdirectory action="create" directory="\wwwroot\images\userimages\#arguments.userid#" />
    </cfif>
    <cfimage name="newimage"
      source="http://ec2url/img/#arguments.imgname#"
      action="write"
      overwrite="yes"
      destination="\wwwroot\images\userimages\#arguments.userid#\#arguments.imgname#">
    <cfreturn true>
  <cfelse>
    <cfreturn false>
  </cfif>
</cffunction>
```

So on the main web server, we have a function that will take a few vars and send them to our EC2 box, the main one being the URL to capture. our EC2 box will save the image and our function knows where it will save it and the file name, so we can just use CFImage to save the file to our local system.

And there you have it.

Quick and simple on a shared host, this is about your only option for doing screen captures that are of good quality.

If there is another way, I would love to hear about it.