

VISUAL BASIC – HOW TO USE VBSCRIPT TO SEND EMAILS WITH GMAIL

I normally don't post scripts and batch files that I use, but I wanted to share this particular script because I use it all the time. **VBScript** files are nice because typically they are simply coded, and work across virtually all **Windows-powered** machines. You just have to throw the code in a **.VBS** file, and from there it can be executed in your batch files very easily.

The contents of the file are below, and once set up you'll be able to send emails using any **Gmail** account straight from the command prompt. In the code there are two things that you need to configure: the email address and password of the account you want to send as. When executing the **VBScript** file you simply pass it three parameters: recipient email address, subject line, and the body of the email. There is an optional **fourth parameter** that it accepts which is the file path to a file you want to attach to the email.

That's it.

Why do I use this all the time? I can easily call this file from other scripts or batch files so that it sends me an email notification whenever something completes. Sometimes I use it just to have a nice history of when some of my batch files complete (**or to verify that they even ran**), but other times I will insert "dynamic" text into the body of the email. For example, if an error occurred I may have it put the error message in the body of the email. An alternative to that would be attaching a log file using the optional fourth parameter.

The nice thing about the way this script works is that you'll be able to send emails without setting up an **SMTP** server on your machine. Of course you won't be able to send out mass quantities of emails since **Google** limits you to around 500 sent messages per day, but that should be more than adequate for personal purposes.

Let's take a look at what the code looks like...

Code begins here:

```
'Usage: cscript sendemail.vbs <email_recipient@example.com> "<subject_line>"
"<email_body>" "<optional:email_attachment_path>"
'Ex. No attach: cscript sendemail.vbs example@gmail.com "test subject line"
"test email body"

'Ex. W/ attach: cscript sendemail.vbs example@gmail.com "test subject line"
"test email body" "c:\scripts\log.txt"

'*****
'****CONFIGURE THE FROM EMAIL ADDRESS AND PASSWORD

Const fromEmail = email\_sender@gmail.com
Const password = "password"

'****END OF CONFIGURATION
'*****

Dim emailObj, emailConfig
Set emailObj = CreateObject("CDO.Message")
```

```
emailObj.From = fromEmail  
emailObj.To = WScript.Arguments.Item(0)  
emailObj.Subject = WScript.Arguments.Item(1)  
emailObj.TextBody = WScript.Arguments.Item(2)
```

```
If WScript.Arguments.Count > 3 Then  
emailObj.AddAttachment WScript.Arguments.Item(3)  
End If
```

```
Set emailConfig = emailObj.Configuration  
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/smtpserver")  
= "smtp.gmail.com"
```

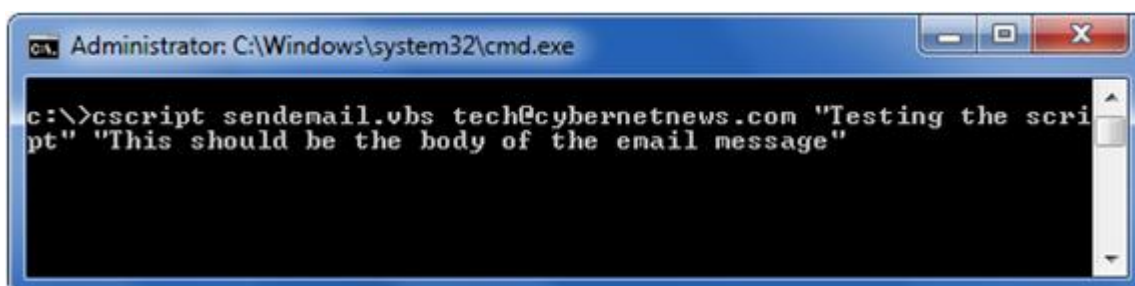
```
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/smtpserverpor  
t") = 465  
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/sendusing") =  
2  
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/smtpauthentic  
ate") = 1
```

```
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/smtpusessl")  
= true  
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/sendusername"  
) = fromEmail  
emailConfig.Fields("http://schemas.microsoft.com/cdo/configuration/sendpassword"  
) = password  
emailConfig.Fields.Update  
emailObj.Send
```

```
Set emailObj = nothing  
Set emailConfig = nothing
```

^ That is the end of the code ([Download the Script](#)) ^

You can see that the code starts with example usage of how you can call the **VBScript** file. Given the nature of executing **VBScript** files you may or may not have to include the "cscript" portion, but in general you are always better off having it there to ensure that it will run on your computer just fine. Here's an example of what it looks like when being executed from the command prompt:



Armed with this script you should be able to take it and throw it into batch files, or call it from anywhere that can execute things via the command line. One thing you may want to consider is creating an extra email account just for sending these emails.

Not only will that keep your "*sent mail*" clean in your primary **Gmail** account, but it will also be a bit more secure since the password for the sender account is stored in plain text within the script.

Why Use This

This script can be used for a lot of different things. it can be also use in other **VBScript** files that when a **task** starts, ends, and any notifications it needs to send in between.