PriveonLabs Research

Cisco Security Agent Protection Series:

CVE-2006-003

Microsoft RDS.Dataspace ActiveX (MDAC) Vulnerability

Zach Brewer
Security Consultant

Overview

The Common Vulnerabilities and Exposures list candidate CVE-2006-003 (MS06-014) details an exploit in the Microsoft Data Access Components. The Microsoft Data Access Components, commonly abbreviated as MDAC, are a collective group of technologies used by programmers to access specific data stores. Common MDAC components include ActiveX Data Objects (ADO) and Open Database Connectivity (ODBC).

In April of 2006, Microsoft reported a vulnerability with the RDS.Dataspace ActiveX control which is distributed via MDAC. RDS.Dataspace is a component of ADO that allows data manipulation between a client and server in a single round trip.

CVE: *2006-003*
IMPACT: *Remote Code Execution*
AFFECTED SOFTWARE:
MS Windows XP SP1 running MDAC 2.7 SP1
MS Windows XP SP2 running MDAC 2.8 SP1
MS Windows XP Professional x64 Edition running MDAC 2.8 SP2
MS Windows Server 2003 running MDAC 2.8
MS Windows Server 2003 SP1 running MDAC 2.8 SP2
MS Windows Server 2003 for Itanium-based Systems running MDAC2.8
MS Windows Server 2003 with SP1 for Itanium-based Systems running MDAC 2.8 SP2
MS Windows Server 2003 x64 Edition running MDAC 2.8 SP2
MS Windows 98, MS Windows 98 (SE), and MS Windows (ME)

Although a vendor issued patch exists for CVE-2006-003 as of this writing, Priveon Incident Response Handlers have seen recent cases of this vulnerability being used to distribute new malware variants. These cases often involve use of the Web Attack Toolkit as a means of vulnerability exploitation and payload delivery. Please see the References section of this document for more information on the Web Attack Toolkit.

The purpose of this document is to explain and expand upon known Proof of Concept (PoC) exploits available as well as how the Cisco Security Agent (CSA) product can protect systems from vulnerabilities that are actively being exploited in the wild.

Exploit Process Overview

CVE-2006-003 is a remotely exploitable vulnerability currently being used to distribute multiple forms of malware, including custom payloads delivered by the malicious web sites and automated attack tools. In some known cases of successful exploitation, email messages are used to encourage victims to visit a malicious website through manipulation and social engineering.

Successful exploitation of the RDS.Dataspace ActiveX control vulnerability could allow a remote attacker to gain full control of the remote system. The remote attacker can, after exploitation, have total control including permissions alteration, software installation, and file manipulation.

PoC Exploit Testing Overview

Testing Environment

The Priveon test lab consisted of several Operating Systems and application combinations including various patch revision levels running on virtual hosts. The purpose of the testing process is to confirm prevention of payload delivery through the Cisco Security Agent.

Cisco Security Agent Version Tested:

- CSA 5.1.0.69 (Default CSA Policies as shown in Figure 1)

Figure 1: Protected Host CSA Policies from v5.1.0.69

Group Name	Version	Description	Policies
<All Windows>		Auto-enrollment group for Windows hosts	2 policies

Policy Name	Version	Description	Rule Modules
Application Classification	5.1 r69	Base policy for behavioral classification of applications.	3 modules
Operating System - Base Permissions - Windows	5.1 r69	Basic permissions for Windows OS	2 modules

Group Name	Version	Description	Policies
Desktops - All types	5.1 r69	Default group for systems that install the Desktop agent kit	9 policies

Policy Name	Version	Description	Rule Modules
Agent UI control	5.1 r69	Policy which governs Agent User Interface	1 module
Document Security - Windows	5.1 r69	Policy to protect user documents	1 module
Email Client - Basic Security - Windows	5.1 r69	Basic application enforcement policy for email client software.	3 modules
General application - Basic Security - Windows	5.1 r69	Basic, Application independent security policy for Windows	3 modules
Installation Applications - Windows	5.1 r69	Software Installers for Windows	4 modules
IP Stack - Internal Network Security	5.1 r69	Policy for protecting the IP Stack on internal systems	1 module
Network Personal Firewall	5.1 r69	Control network access and provide some end user access controls.	1 module
Operating System - Base Protection - Windows	5.1 r69	Basic protection for Windows OS	6 modules
Virus Scanner - Windows	5.1 r69	Application enforcement policy for virus scanner software.	1 module

Exploit PoC Testing Results – Unprotected System

The testing process involved Proof of Concept code obtained from MetaSploit.com as part of the MetaSploit framework version 2.7. Various payloads were utilized during the testing process however only the following two are included in the writing of this document: win32_reverse and win32_vncinject. The lab virtual host was run in CSA TestMode in order to observe results of the exploit process for each respective payload.

Win32_Reverse Payload

The win32_reverse payload is commonly used to demonstrate payloads capable of traversing network firewalls and Access Control Lists (ACLs). Many organizations have strict inbound Access Control Lists on ingress firewalls but much more lenient outbound access policies. Win32_reverse and similar payloads are particularly effective when an attacker uses a port that is commonly allowed outbound connectivity such as TCP/80.

The win32_reverse payload requires the victim to connect to a malicious web page for payload delivery as seen in figure 2. Once delivered, the payload connects back to a specified host and gives the attacker full control of the system via a remote command shell as demonstrated in figure 3. This exploit process and type of payload can often circumvent normal network access control lists and firewall policies that an organization may have in place.

Figure 2: Victim visiting a malicious website that delivers win32_reverse

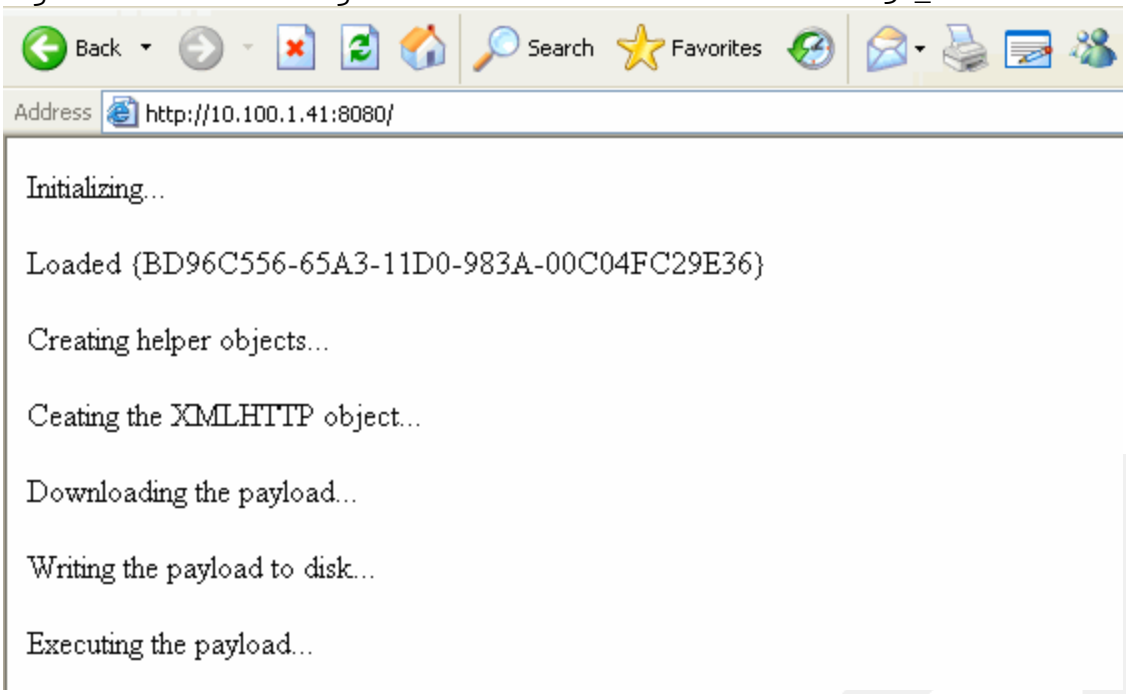
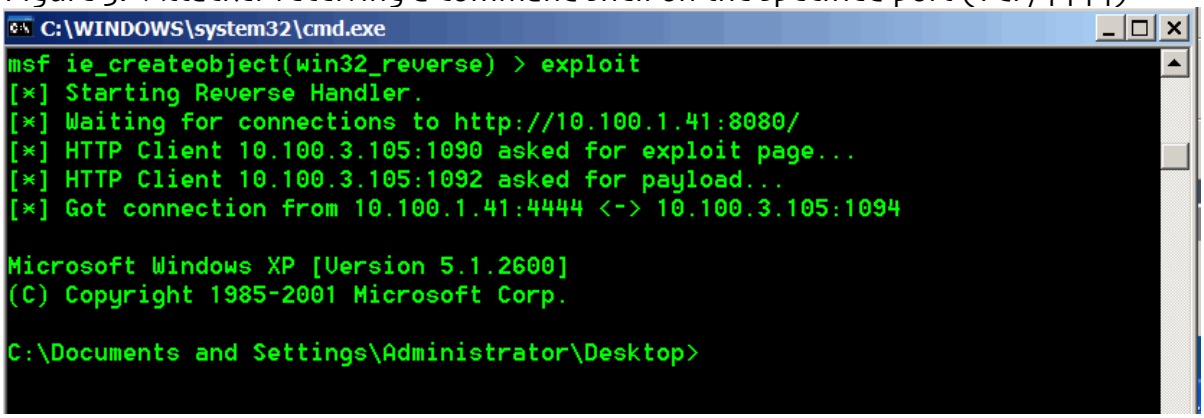


Figure 3: Attacker receiving a command shell on the specified port (TCP/4444)



Win32_Reverse_VNC

The win32_reverse_vnc payload is an example of an advanced staged payload. After successful exploitation, the win32_revese_vnc payload loads a custom VNC DLL and initiates a reverse VNC connection back to the attacker as seen in figure 4. The result is that the attacker has full control of the victim through means of a remote VNC session as seen in figure 4.

Figure 4: MetaSploit sending the Win32_Reverse_VNCInject exploit

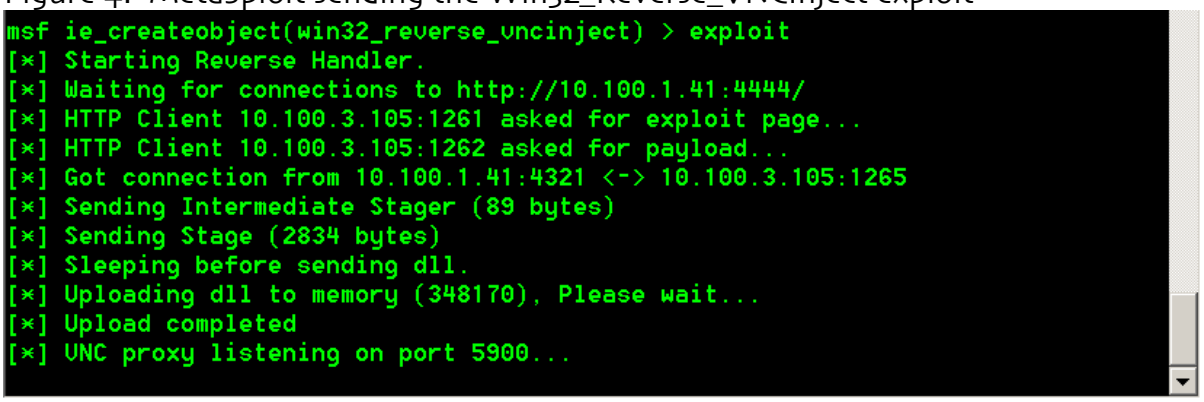
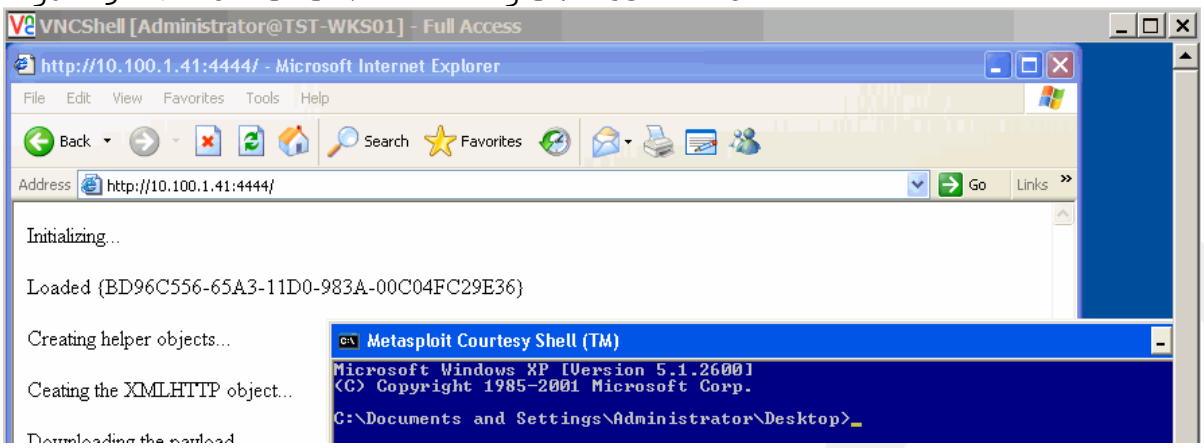


Figure 5: Remote attacker receiving a VNC Shell from the victim.



CSA Protected Exploit PoC Testing Results

After testing the selected payloads and verifying that they were successful when used with the MetaSploit PoC code for CVE-2006-003, the payloads were used with the Cisco Security Agent installed on the lab virtual hosts.

CSA successfully prevented the win32_reverse payload from delivery through a series of File Access Control and System API rules as seen in figure 6. Again, we must stress that the lab system was placed in CSA TestMode in order to identify each of the rules that were triggered at different stages of the exploit process. The first two rules that triggered were file access control rules that prevented applications executing untrusted content from accessing protected system resources.

NOTE: You may notice that some events reference CSA user query messages. CSA rules that utilize the Query rule action are dependant upon the level of interactivity provided by the CSA administrator. If certain elements of the CSA User Interface (UI) are disabled by the administrator, then the rules take the default action as specified in the given rule (most often DENY).

Figure 6: CSA events with Win32_Reverse payload

#	Date	Host	Severity	Event
12	11/21/2006 10:27:34 AM	TST-WKS01	Alert	TESTMODE: The current application 'C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe' (as user TST-WKS01\Administrator) attempted to execute the new application 'C:\WINDOWS\system32\cmd.exe'. The operation would have been denied. Details Rule 464 Wizard Find Similar
11	11/21/2006 10:27:34 AM	TST-WKS01	Notice	TESTMODE: The process 'C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'The process C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe is attempting to invoke a system function from a buffer. Do you wish to allow this?' Details Rule 182 Wizard Find Similar
10	11/21/2006 10:27:34 AM	TST-WKS01	Notice	TESTMODE: The process 'C:\Program Files\Internet Explorer\IEXPLORE.EXE' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'A process is attempting to invoke C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe which has been recently downloaded and may be dangerous. Do you wish to allow this?' Details Rule 457 Wizard Find Similar
9	11/21/2006 10:27:32 AM	TST-WKS01	Notice	TESTMODE: The process 'C:\Program Files\Internet Explorer\IEXPLORE.EXE' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'The downloaded script C:\Program Files\Internet Explorer\IEXPLORE.EXE is trying to access system resources. This is potentially dangerous. Do you wish to allow this?' Details Rule 177 Wizard Find Similar

The win32_reverse_vncinject payload provided similar results as seen in figure 7. This is to be expected since MetaSploit uses a similar process for the initial steps in payload delivery for many vulnerabilities.

Figure 7: CSA events with Win32_Reverse_VNCInject payload

#	Date	Host	Severity	Event
22	11/21/2006 12:52:33 PM	TST-WKS01	Alert	TESTMODE: The current application 'C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe' (as user TST-WKS01\Administrator) attempted to execute the new application 'C:\WINDOWS\system32\cmd.exe'. The operation would have been denied. Details Rule 464 Wizard Find Similar
21	11/21/2006 12:52:24 PM	TST-WKS01	Notice	TESTMODE: The process 'C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'The process C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe is attempting to invoke a system function from a buffer. Do you wish to allow this?' Details Rule 182 Wizard Find Similar
20	11/21/2006 12:52:20 PM	TST-WKS01	Notice	TESTMODE: The process 'C:\Program Files\Internet Explorer\IEXPLORE.EXE' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'A process is attempting to invoke C:\Documents and Settings\Administrator\Local Settings\Tempmetasploit.exe which has been recently downloaded and may be dangerous. Do you wish to allow this?' Details Rule 457 Wizard Find Similar
19	11/21/2006 12:52:17 PM	TST-WKS01	Notice	TESTMODE: The process 'C:\Program Files\Internet Explorer\IEXPLORE.EXE' (as user TST-WKS01\Administrator) attempted to access a resource which would have resulted in the user being asked the following question. 'The downloaded script C:\Program Files\Internet Explorer\IEXPLORE.EXE is trying to access system resources. This is potentially dangerous. Do you wish to allow this?' Details Rule 177 Wizard Find Similar

Summary of Results

CSA successfully demonstrated the ability to prevent delivery of the several payloads tested with the Microsoft RDS.Dataspace ActiveX vulnerability. Through a combination of File Access Control Rules, System API Rules, and Dynamic Content Classification Policies, CSA was able to successfully detect and prevent the attempted attack(s) at various stages in payload delivery.

One consideration CSA administrators should heed is the level of agent interactivity (agent UI access) provided to users. Query rules observed in the initial phases of the exploit process (with agent interactivity) depend directly on the user to prevent the first steps of the exploit process. It should be noted, however, that thanks to CSA's defense-in-depth approach to security, the exploit would have been caught with all tested payloads regardless of user interactivity.

It is important to understand the impact of this vulnerability on your network and also the possibility of exploitation. ActiveX controls should be of specific concern because of the attacker's capability to force a remote user to load previous vulnerable version of the controls without user knowledge if the user had previously allowed their system to always trust Microsoft signed content.

As demonstrated with the Microsoft RDS.Dataspace ActiveX vulnerability, when used as a part of an overall layered security approach, CSA is a valuable tool against both known and unknown security threats.

References

- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2006-003
- http://metasploit.com
- http://www.websense.com/securitylabs/alerts/alert.php?AlertID=472
    - Includes Web-Attacker Toolkit Related Information

Where to Go for More Information:

Custom Research Documents
Exploit Reverse Engineering/Forensics
Security Implementation
Real-World Training
Managed Services

Available @ www.priveon.com