



[All Products](#) | [Support](#) | [Search](#) | [microsoft.com Home](#)



[TechNet Home](#) | [Downloads](#) | [Site Map](#) | [TechNet Worldwide](#) |

Search [Advanced Search](#)

 All TechNet  Knowledge Base

sync toc  

- TechNet Home
- Products & Technologies
- IT Tasks
- IT Solutions
- Script Center
- Security
- Downloads
- How-To Index
- Columns
- Troubleshooting & Support
- IT Community
- Training, Career & Events
- Subscriptions
- Using TechNet
- TechNet Archive

I want to:

GO

Microsoft Security Bulletin MS02-065



Buffer Overrun in Microsoft Data Access Components Could Lead to Code Execution (Q329414)

Originally posted: November 20, 2002

Summary

Who should read this bulletin: Customers using Microsoft® Windows®, particularly those who operate web sites or browse the Internet.

Impact of vulnerability: Run code of attacker's choice

Maximum Severity Rating: Critical

Recommendation: Users should apply the patch immediately.

Affected Software:

- Microsoft Data Access Components (MDAC) 2.1
- Microsoft Data Access Components (MDAC) 2.5
- Microsoft Data Access Components (MDAC) 2.6
- Microsoft Internet Explorer 5.01
- Microsoft Internet Explorer 5.5
- Microsoft Internet Explorer 6.0

Note: The vulnerability does not affect Windows XP, despite the fact that it uses Internet Explorer 6.0. Windows XP customers do not need to take any action.

End User Bulletin: An end user version of this bulletin is available at:

http://www.microsoft.com/security/security_bulletins/ms02-065.asp

Technical details

Technical description:

Microsoft Data Access Components (MDAC) is a collection of components used to provide database connectivity on Windows platforms. MDAC is a ubiquitous technology, and it is likely to be present on most Windows systems:

- It is included by default as part of Windows XP, Windows 2000, and Windows Millennium.
- It is available for download as a stand-alone technology in its own right
- It is either included in or installed by a number of other products and technologies. For instance, MDAC is included in the Windows NT® 4.0 Option Pack, and some MDAC components are present as part of Internet Explorer even if MDAC itself is not installed.

MDAC provides the underlying functionality for a number of database operations, such as connecting to remote databases and returning data to a client. One of the MDAC components, known as Remote Data Services (RDS), provides functionality that support three-tiered architectures – that is, architectures in which a client's requests for service from a back-end database are intermediated through a web site that applies business logic to them. A security vulnerability is present in the RDS implementation, specifically, in a function called the RDS Data Stub, whose purpose it is to parse incoming HTTP requests and generate RDS commands.

A security vulnerability resulting from an unchecked buffer in the Data Stub affects versions of MDAC prior to version 2.7 (the version that shipped with Windows XP). By sending a specially malformed HTTP request to the Data Stub, an attacker could cause data of his or her choice to overrun onto the heap. Although heap overruns are typically more difficult to exploit than the more-common stack overrun, Microsoft has confirmed that in this case it would be possible to exploit the vulnerability to run code of the attacker's choice on the user's system.

Both web servers and web clients are at risk from the vulnerability:

- Web servers are at risk if a vulnerable version of MDAC is installed and running on the server. To exploit the vulnerability against such a web server, an attacker would need to establish a connection with the server and then send a specially malformed HTTP request to it, that would have the effect of overrunning the buffer with the attacker's chosen data. The code would run in the security context of the IIS service (which, by default, runs in the LocalSystem context)
- Web clients are at risk in almost every case, as the RDS Data Stub is included with all current versions of Internet Explorer and there is no option to disable it. To exploit the vulnerability against a client, an attacker would need to host a web page that, when opened, would send an HTTP reply to the user's system and overrun the buffer with the attacker's chosen data. The web page could be hosted on a web site or sent directly to users as an HTML Mail. The code would run in the security context of the user.

Clearly, this vulnerability is very serious, and Microsoft recommends that all customers whose systems could be affected by them take appropriate action immediately.

- Customers using Windows XP, or who have installed [MDAC 2.7](#) on their systems are at no risk and do not need to take any action.
- Web server administrators who are running an affected version of MDAC should either install the patch, disable MDAC and/or RDS, or upgrade to [MDAC 2.7](#), which is not affected by the vulnerability.
- Web client users who are running an affected version of MDAC should install the patch immediately on any system that is used for web browsing. It is important to stress that the latter guidance applies to any system used for web browsing, regardless of any other protective measures that have already been taken. For instance, a web server on which RDS had been disabled would still need the patch if it was occasionally used as a web client.

Before deploying the patch, customers should familiarize themselves with the caveats discussed in the FAQ and in the Caveats section below.

Mitigating factors:

Web Servers

- Web servers that are using [MDAC version 2.7](#) (the version that shipped with Windows XP) or later are not at risk from the vulnerability.
- Even if a vulnerable version of MDAC were installed, a web server would only be at risk if RDS were enabled. RDS is disabled by default on clean installations of Windows XP and Windows 2000, and can be disabled on other systems by following the guidance in the [IIS Security Checklist](#). In addition, the [IIS Lockdown Tool](#) will automatically disable RDS when used in its default configuration.
- If the [URLScan](#) tool were deployed with its default ruleset (which allows only ASCII data to be present in an HTTP request), it is likely that the vulnerability could only be used for denial of service attacks.
- IIS can be configured to run with fewer than administrative privileges. If this has been done, it would likewise limit the privileges that an attacker could gain through the vulnerability.
- IP address restrictions, if applied to the RDS virtual directory, could enable the administrator to restrict access to only trusted users. This is, however, not practical for most web server scenarios.

Web clients

- Web clients that are using [MDAC version 2.7](#) (the version that shipped with Windows XP) or later are not at risk from the vulnerability.
- The HTML mail-based attack vector could not be exploited automatically on systems where Outlook 98 or Outlook 2000 were used in conjunction with the [Outlook Email Security Update](#), or Outlook Express 6

or Outlook 2002 were used in their default configurations.

- Exploiting the vulnerability would convey to the attacker only the user's privileges on the system. Users whose accounts are configured to have few privileges on the system would be at less risk than ones who operate with administrative privileges.

Severity Rating:

MDAC 2.1	Critical
MDAC 2.5	Critical
MDAC 2.6	Critical
MDAC 2.7	Not affected
Internet Explorer 5.01	Critical
Internet Explorer 5.5	Critical
Internet Explorer 6.0	Critical

The above [assessment](#) is based on the types of systems affected by the vulnerability, their typical deployment patterns, and the effect that exploiting the vulnerability would have on them. This vulnerability is rated critical because an attacker could take over an IIS server or an Internet Explorer client and run code. Any IIS server with MDAC and all Internet Explorer clients should apply the patch immediately.

Vulnerability identifier: [CAN-2002-1142](#)

Tested Versions:

Microsoft tested MDAC 2.1, 2.5, 2.6 and 2.7 to assess whether they are affected by the server-side vulnerability. In addition, Microsoft also tested Internet Explorer 5.01, 5.5 and 6.0 to assess whether they are affected by the client-side vulnerability. Previous versions are no longer [supported](#), and may or may not be affected by these vulnerabilities.

Frequently asked questions

What vulnerability is eliminated by the patch?

This patch eliminates a [security vulnerability](#) affecting many web servers and clients. However, before installing it, it's worth reviewing an important [caveat](#) associated with the patch.

What caveats are associated with the patch?

Although the patch does address the vulnerability, there is a niche scenario through which a patched system could, under unusual conditions, be made vulnerable again. This scenario results because it is not possible to set the "Kill Bit" used by one of the vulnerable components.

What's the "Kill Bit"?

The Kill Bit is a method by which an ActiveX control can be prevented from ever being invoked via Internet Explorer, even if it's present on the system. (More information on the Kill Bit is available in Microsoft Knowledge Base article Q240797). Typically, when a security vulnerability involves an ActiveX control, the patch delivers a new control and sets the Kill Bit on the vulnerable control. However, it isn't feasible to do so in this case.

Why isn't it feasible to set the Kill Bit in this case?

The ActiveX control involved in these vulnerabilities is used in many applications and web pages to access data. Many applications, including third-party applications, contain hard-coded references to it; if the patch set the Kill Bit, the web pages would no longer function at all – even with the new, corrected version. As a result, the patch updates the control to remove the vulnerabilities, but does not provide a brand-new control and set the Kill Bit on the old one.

What the risk associated with taking this approach?

Because the ActiveX control at issue here has been digitally signed by Microsoft, and the signature is still valid, it could be possible under certain conditions for an attacker to re-introduce the old, vulnerable version of the control onto a system that had been patched, thereby making it vulnerable again. In order for this happen, though, the user would need to either visit a web site operated by a malicious person or open an HTML mail from one. (It's worth noting that in the case of HTML mail, customers using either Outlook 6 or Outlook 2002 in the default configuration, or Outlook 98 or 2000 in conjunction with the [Outlook Email Security Update](#), would be at no risk).

Why would an attacker be able to silently re-introduce the old version of the control? Shouldn't there be a warning message?

A warning message is generated anytime there's an error associated with a digital signature (e.g., a bad signature or expired certificate) or the signer isn't trusted. But in this case, the digital signature on the old version of the control is still valid, and the signer is Microsoft – which is a trusted publisher in many cases. Because of this, most users would not see a warning message of any kind if the old control was re-introduced.

Why not revoke the certificate that was used to sign the control?

The certificate that was used to sign the control is still valid – the problem lies in the control, not the certificate. In addition, a number of controls have been signed using the same certificate, and revoking the certificate would cause all of them to become invalid.

What steps could I follow to prevent the control from being silently re-introduced onto my system?

The simplest way is to make sure you have no trusted publishers, including Microsoft. If you do that, any attempt by either a web page or an HTML mail to download an ActiveX control will generate a warning message. Here's how to empty the Trusted Publishers list:

1. In Internet Explorer, choose Tools, then Internet Options.
2. Select the Content tab. In the Certificates section of the page, click on Publishers.
3. In the Certificates dialog, click on the Trusted Publishers tab.
4. For each certificate in the list, click on the certificate and then select Remove. Confirm that you want to remove the entry.
5. When you've removed all entries from the list, select Close to close the Certificates dialog, then click on OK to close the Internet Options dialog.

After emptying the Trusted Publishers list, if I do see a warning saying that a web site or an HTML mail wants to download a control, how can I decide whether to let it proceed?

The best criterion to use is whether you trust the web site or the sender of the HTML mail. If you don't trust the web site offering the control, cancel the download.

Will Microsoft eventually set the Kill Bit on this control?

Yes. Microsoft is developing a new technology that will enable it to set the Kill Bit on the vulnerable version of the control without forcing users to re-author web pages containing references to these controls. When the new technology is available, we will ensure that this fix uses it.

What's the scope of the vulnerability?

This is a [buffer overrun](#) vulnerability. An attacker who successfully exploited it could gain complete control over an affected system, thereby gaining the ability to take any action that the legitimate user could take. This could include creating, modifying or deleting data on the system, reconfiguring it, reformatting the hard drive, or running programs of the attacker's choice on it. The vulnerability poses a risk both to web servers and web clients, and Microsoft recommends that all users take action immediately to ensure that their systems are protected against it.

Windows XP systems, whether serving as Web servers or clients, are not affected by the vulnerability. Other systems have varying degrees of options available to them:

- Web servers have a range of actions they can take to protect their systems, from installing the patch to disabling the affected function. Even in cases where a server is vulnerable, tools such as URLScan would likely limit the use of the vulnerability to denial of service attacks only.
- Web clients – including Web servers that are sometimes used for Web browsing – have fewer options. Web clients running anything but Windows XP can only be made secure by applying the patch.

What causes the vulnerability

The vulnerability results because of an unchecked buffer in one of the Microsoft Data Access Components – specifically, the Remote Access Service Data Stub.

What are the Microsoft Data Access Components?

[Microsoft Data Access Components](#) (MDAC) is a collection of components that make it easy for programs to access databases and manipulate the data within them. Modern databases may take a variety of forms (e.g., SQL databases, Access databases, XML files, and so forth) and be housed in a variety of locations (e.g., on the local system or on a remote database server). MDAC provides a consolidated set of functions for working with all of them in a consistent manner.

Do I have MDAC on my system?

The answer is almost certainly yes. MDAC is a ubiquitous technology:

- It installs as part of Windows XP, Windows Me, Windows 2000. (It's worth noting, though, that the version installed by Windows XP does not have this vulnerability)
- It's available for download from the [Microsoft web site](#).
- It's installed by many other Microsoft applications. To name just a few cases, it's installed as part of the Windows NT 4.0 Option Pack and by both Microsoft Access and SQL Server.
- Some of the components in MDAC are included in other Microsoft technologies. For instance, Internet Explorer includes some MDAC functions. As we'll discuss later, this turns out to be an important factor in this case.

What are the Remote Data Services?

Remote Data Services ([RDS](#)) is a component of MDAC. RDS provides a function that's frequently needed in Internet-based scenarios, namely, the ability to access data sources indirectly through a three-tiered system. If you've ever visited a web site that implements a search function, you've participated in a [three-tiered](#) database system. In such a system, the user (who occupies the so-called User Interface Tier) interrogates a database (which occupies the so-called Database Tier), but doesn't do so directly. Instead, he or she provides requests to an intermediary tier, known as the Business Logic Tier. In most Internet scenarios, the Business Logic Tier resides on a web server.

The purpose of the Business Logic Tier is to determine what the user wants, translate that request into a series of database commands, check those commands to ensure that the user is really allowed to make them, and then send them to the Database Tier. When the response from the database arrives, the Business Logic Tier may need to translate the results into a form that's more meaningful for the user. RDS provides many of the functions needed to implement the Business Logic Tier of such a system; specifically, it provides functions that, on a web server, allow it to interpret database requests from a client and, on a client, interpret responses to such requests when they're received from the web server.

What's wrong with RDS?

One of the components of RDS that was delivered in MDAC 2.4, 2.5 and 2.6 contains an unchecked buffer. On the server side, this component is known as the RDS Data Stub and on the client side it is called the Data Space control. These components implement some of the functionality of the Business Logic Tier. In particular, the Data Stub processes HTTP requests and transforms them into RDS requests that can then be passed to the RDS core functionality for processing.

What do you mean when you say that the RDS Data Stub has an unchecked buffer?

A buffer is a location in memory that's allocated to hold data of some type. It's the responsibility of the program that owns the buffer (in this case, the RDS Data Stub) to ensure that it never puts more data into the buffer than it can hold – otherwise, the data will spill into surrounding memory and overwrite the data there, resulting in a buffer overrun.

Buffer overruns are dangerous. In the least serious case, if a buffer were overrun with random data, it would have the effect of corrupting the memory that it overran; in most cases, this would lead to the program, or potentially the system itself, failing. But if the buffer were overrun with carefully selected data, the effect could be to, in essence, alter the program so that it now performed new functions. Clearly, any flaw that could enable an attacker to turn an already running program to his or her own purposes is serious.

What would this vulnerability enable an attacker to do?

This vulnerability would enable an attacker to send an HTTP request to an affected system that, when processed by the RDS Data Stub, would cause a buffer overrun. Potentially, any system that has MDAC, and in particular RDS, installed and running is at risk. But two types of systems are at special risk:

- Web servers. Many web servers have vulnerable versions of RDS running on them. If an attacker successfully exploited the vulnerability against such a server, he or she could either destabilize it or, in the worst case, gain complete control over it. The attacker could then deface web pages, attack users who subsequently visited the site, or simply reformat the hard drive.
- Web clients. By a web client, we mean any system that's used to process web pages; typical examples include home computers, laptops or workstations that are used to browse the Web or handle email. The RDS Data Stub is present on these systems as part of Internet Explorer. If an attacker successfully exploited the vulnerability against such a system, he or she could either cause Internet Explorer fail (which would have no lasting effects) or, in the worst case, gain the user's privileges on it. The attacker could then take any action on the system that the user could take.

Who could exploit the vulnerability?

It would depend on whether the attacker wanted to exploit the vulnerability against a web server or a web client.

- Web server. Any user who could establish a web session with an affected server could exploit the vulnerability, by sending it an appropriate HTTP request.
- Web client. A user could exploit the vulnerability against a web client if he or she were able to construct a web page that would send an appropriate HTTP command, and then convince a user to open it. Typically, this would be done by either hosting the page on a web site that the attacker controlled or sending it directly to users as an HTML mail.

I run a web server. How can I tell whether my system is at risk?

In order for a web server to be at risk from the vulnerability, both of the following must be true:

- A vulnerable version of MDAC must be installed on the server. The most recent version of MDAC, [version 2.7](#) (which ships as part of Windows XP), does not contain this vulnerability. However, most previous versions are vulnerable.
- RDS must be running in Internet Information Services (IIS). In IIS 5.0 and 5.1, RDS is disabled by default (unless the system was upgraded from a previous version of Windows). Even in cases where RDS does run by default, it can be disabled as discussed in the [IIS Security Checklist](#). The [IIS Lockdown Tool](#) will also automatically disable RDS when used in its default configuration.

It's important to keep in mind, though, that if the web server is also used as a web client occasionally (that is, if you browse the web or read email from the server), it could still be at risk. The server-based and client-based vulnerabilities are completely independent of one another.

I've installed the URLScan tool on my web server. Will it help protect my system?

Yes. The [URLScan](#) tool restricts the type of HTTP requests that the server will process. Of particular interest in this case is the fact that URLScan's default ruleset will only allow HTTP requests to be processed by the server if they consist of only ASCII data. It would be extremely difficult to create a request that would alter the operation of the IIS service using only valid ASCII data; however, even in this case, an attacker could still cause the service to fail.

My system is a web client. How can I tell if it's at risk?

The first thing to do is check whether you're running Windows XP. If you are, your system is at no risk – the version of MDAC that shipped with Windows XP does not contain the vulnerability.

All other versions of Windows are at risk. Several versions of Windows ship with a vulnerable version of MDAC, as did several versions of Internet Explorer. As a result, systems running anything other than Windows XP are almost certainly at risk and need the patch.

You said that Windows XP isn't vulnerable, but that customers using Internet Explorer 6.0 are. Yet Internet Explorer 6.0 shipped as part of Windows XP. Why isn't Windows XP vulnerable?

When Internet Explorer 6.0 is installed on a system, it checks to see whether there's a version of MDAC already installed; if there isn't one, it installs it. In the case of Windows XP, a version of MDAC is already installed – one that isn't affected by the vulnerability – and so Internet Explorer 6.0 uses that version.

Does the web site-based or HTML mail-based attack vector pose the greater threat to web clients?

There would be advantages and disadvantages for the attacker regardless of the attack vector chosen. The primary advantage, from the attacker's perspective, of hosting the web page on a web site is that most computers would be vulnerable to such an attack unless the patch had been installed. The primary disadvantage is that the attacker wouldn't have any way to force users to visit the site. Instead, he or she would need to lure them there, typically by getting them to click a link that would take them to the attacker's site.

In contrast, sending the web page as an HTML mail would offer the attacker the advantage of being able to target specific users, and send it directly to them. The primary disadvantage is that the HTML mail-based attack would fail on many users' systems. Specifically, even without the patch, the vulnerability could not be exploited via HTML mail on systems where Outlook 98 or Outlook 2000 were used in conjunction with the [Outlook Email Security Update](#), or Outlook Express 6 or Outlook 2002 were used in their default configurations.

My system is a web server, and I've confirmed that it's not vulnerable. However, I also sometimes browse the Web from that system. Do I need to install the patch?

Yes. Any system that acts as a web client needs the patch. This is true even if the system also happens to be a web server, and even if the web server has been configured in a way to protect it from the vulnerability.

Is there a separate patch for MDAC and Internet Explorer?

No. We have developed a single patch that will install the fixes for both MDAC and Internet Explorer at the same time. The patch will determine what version of MDAC, if any, your system is using and apply the fixes to all vulnerable components on it. If there are no vulnerable components on the system, the patch will do nothing.

I don't know if MDAC is installed. Do I need to determine that first before I apply the patch?

No. The patch will determine what version of MDAC, if any, is installed on your system and apply the needed fixes.

Patch availability

Download locations for this patch

- The following patch can be installed on all affected platforms:
<http://www.microsoft.com/downloads/Release.asp?ReleaseID=44733>

Additional information about this patch

Installation platforms:

The patch can be installed on the following systems:

- Windows 98 Gold.
- Windows 98SE Gold
- Windows Me Gold
- Windows NT4 [Service Pack 6a](#)
- Windows 2000 [Service Pack 2](#) or [Service Pack 3](#)

Inclusion in future service packs:

- The fix for this issue will be included in the next service pack for MDAC 2.5. There will be no more service packs for MDAC 2.1 and MDAC 2.6.
- The fix will also be included in Internet Explorer 5.01 Service Pack 4 and Internet Explorer 6.0 Service Pack 2.

Reboot needed:

- Web servers: We recommend rebooting the server after installing the patch.
- Web client: It is not necessary to reboot after installing the patch.

Patch can be uninstalled: No.

Superseded patches: None.

Verifying patch installation:

- Microsoft Knowledge Base article [Q329414](#) provides a file manifest that can be used to verify the patch installation.

Caveats:

- As discussed in the FAQ, the patch does not set the Kill Bit on the affected ActiveX control.
- If, after applying the patch, an MDAC service pack that predates the patch is installed, the effect is to remove the patch. Moreover, because the patch files would still be on the system, Windows Update would not be able to detect that the patch files were not in use, and would not offer to reinstall the patch. Instead, the user would need to reinstall the patch manually after installing the service pack.

An example would be a users who have already patched their MDAC 2.5 machines. Then if they apply

MDAC 2.5 Service Pack 2 over the already patched MDAC 2.5 machines, it's possible that there would be a regression, making it necessary for the users to reinstall this patch.

Users can always upgrade to the latest version of MDAC since [MDAC 2.7](#) is not affected by this vulnerability.

Localization:

Localized versions of this patch are available at the locations discussed in "Patch Availability".

Obtaining other security patches:

Patches for other security issues are available from the following locations:

- Security patches are available from the [Microsoft Download Center](#), and can be most easily found by doing a keyword search for "security_patch".
- Patches for consumer platforms are available from the [WindowsUpdate](#) web site

Other information:

Acknowledgments

Microsoft [thanks Foundstone Research Labs](#) for reporting this issue to us and working with us to protect customers.

Support:

- Microsoft Knowledge Base article [Q329414](#) discusses this issue. Knowledge Base articles can be found on the [Microsoft Online Support](#) web site.
- Technical support is available from [Microsoft Product Support Services](#). There is no charge for support calls associated with security patches.

Security Resources: The [Microsoft TechNet Security](#) Web Site provides additional information about security in Microsoft products.

Disclaimer:

The information provided in the Microsoft Knowledge Base is provided "as is" without warranty of any kind. Microsoft disclaims all warranties, either express or implied, including the warranties of merchantability and fitness for a particular purpose. In no event shall Microsoft Corporation or its suppliers be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages, even if Microsoft Corporation or its suppliers have been advised of the possibility of such damages. Some states do not allow the exclusion or limitation of liability for consequential or incidental damages so the foregoing limitation may not apply.

Revisions:

- V1.0 (November 20, 2002): Bulletin Created.

[Contact Us](#) | [E-mail this Page](#) | [TechNet Newsletter](#)

© 2002 Microsoft Corporation. All rights reserved. [Terms of Use](#) [Privacy Statement](#) [Accessibility](#)