

# [NT] Malformed RPC Request Can Cause Service Failure (Exchange, SQL, Windows)

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2001-08/0013.html>

---

*From:* [support@securiteam.com](mailto:support@securiteam.com)

*Date:* 07/30/01

From: [support@securiteam.com](mailto:support@securiteam.com)

To: [list@securiteam.com](mailto:list@securiteam.com)

Subject: [NT] Malformed RPC Request Can Cause Service Failure (Exchange, SQL, Windows)

Message-Id: <20010730051334.6A60F138BF@mail.der-keiler.de>

Date: Mon, 30 Jul 2001 07:13:34 +0200 (CEST)

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

Malformed RPC Request Can Cause Service Failure (Exchange, SQL, Windows)

---

## SUMMARY

Several of the RPC servers associated with system services in Microsoft Exchange, SQL Server, Windows NT 4.0 and Windows 2000 do not adequately validate inputs, and in some cases will accept invalid inputs that prevent normal processing. The specific input values at issue here vary from RPC server to RPC server.

An attacker who sent such inputs to an affected RPC server could disrupt its service. The precise type of disruption would depend on the specific service, but could range in effect from minor (e.g., the service temporarily hanging) to major (e.g., the service failing in a way that would require the entire system to be restarted).

## DETAILS

Affected software:

- \* Microsoft Exchange Server 5.5
- \* Microsoft Exchange Server 2000
- \* Microsoft SQL Server 7.0
- \* Microsoft SQL Server 2000
- \* Microsoft Windows NT 4.0
- \* Microsoft Windows 2000

Mitigating factors:

Proper firewalling would help minimize an affected system's exposure to

## Hypermail Development List: [NT] Malformed RPC Request Can Caus

attack by Internet-based users. In general, a firewall should block access to all RPC services except those that are specifically intended for use by untrusted users.

Patch availability:

Download locations for this patch

\* Microsoft Exchange Server 5.5:

<<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31517>>

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31517>

\* Microsoft Exchange Server 2000:

<<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31522>>

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31522>

Also included in Exchange Server 2000 Service Pack 1.

\* Microsoft SQL Server 7.0:

<<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31645>>

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31645>

Also included in SQL Server 7.0 Service Pack 3.

\* SQL Server 2000:

<<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31644>>

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31644>

Also included in SQL Server 2000 Service Pack 1.

\* Microsoft Windows NT 4.0 Workstation, Windows NT 4.0 Server, and Windows NT 4.0 Server, Enterprise Edition:

Included in the Windows NT 4.0 Security Roll-up

\* Microsoft Windows NT 4.0 Server, Terminal Server Edition:

Will be included in the Windows NT 4.0 Security Roll-up for Terminal Server (to be released shortly).

\* Microsoft Windows 2000 Professional, Server and Advanced Server:

<<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31434>>

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=31434>

\* Microsoft Windows 2000 Datacenter Server:

Patches for Windows 2000 Datacenter Server are hardware-specific and available from the original equipment manufacturer.

What's the scope of this vulnerability?

This is a denial of service vulnerability. By sending a specially malformed request to an affected system, an attacker could disrupt its ability to service legitimate users' requests.

The effect of exploiting this vulnerability would vary widely, depending on the particular request, and which of the affected services the attacker could send it to. If best practices have been followed, an attacker on the Internet would be unable to send such a request to any of the affected services.

What causes the vulnerability?

The vulnerability results because of mismatches between the interface definitions in several RPC server stubs and the input validation code in the associated servers. In the affected servers, certain inputs are not validated prior to use, with the result that inputs that are permissible per the interface definitions but which nevertheless are invalid could be

## Hypermail Development List: [NT] Malformed RPC Request Can Caus

used to disrupt the server operation.

What is RPC?

RPC (Remote Procedure Call) is a technology that is used extensively to support distributed applications — that is, applications whose various components are located on different computers. The primary purpose of RPC is to provide a way for the components to communicate with each other. This allows the components to levy requests on each other and communicate the results of these requests.

What's an RPC server stub?

The overall goal of RPC is to mask the fact that the client and server components reside on different machines, and instead make it appear that both are running on the local machine. This is accomplished with stubs. On the client system, a stub (known as the client stub) makes it appear that the server component resides on the client machine. Likewise, on the server system, a stub (known as the server stub) makes it appear that the client component resides on the server machine.

When the client component levies a request to the client stub, the stub packages the request in an RPC message and sends it to the server machine. The server stub unpacks the request and passes it to the server component, which acts on the request. If the server needs to send a response, it sends it to the server stub, which then packages the response in an RPC message and sends it to the client machine. The client stub then unpacks the response and passes it to the client component.

What is wrong with RPC?

The problem does not lie in the RPC architecture, but rather in the implementation of several RPC servers. The server stub advertises an interface definition, but the servers do not always validate the inputs they receive correctly.

What is an interface definition?

An interface definition can be thought of as a template that all requests to a particular server must conform to. For instance, an interface definition for a particular RPC server might indicate that there are five parameters that must be included in a request, and that all of them must be integers. A request that does not adhere to the interface definition will not be accepted by the server stub.

The problem is that even a request that conforms to the interface definition may not be valid. For instance, even though an interface definition may require an integer as an input, there may be values that the server code cannot process. It is the responsibility of the server code to check all inputs to make sure they have acceptable values. This vulnerability results because some RPC servers associated with system services in Exchange, SQL, Windows NT 4.0, and Windows 2000 do not do this.

## Hypermail Development List: [NT] Malformed RPC Request Can Caus

What would be the result if someone sent an invalid request to such a server?

It would depend on the specific server at issue, and how it handles the specific request included in the RPC message. In some cases, the request might have little or no lasting effect on the system service. In others, the request could cause the service could fail with no effect on the overall system. In others, the service could fail in a way that destabilizes the overall system and requires the machine to be restarted.

How might an attacker exploit this vulnerability?

As a general statement, an attacker might exploit this vulnerability as a means of preventing the server from providing useful service. However, the specific effects he could cause would vary dramatically on a machine-by-machine basis. As we discussed above, different services are affected in different ways by this vulnerability.

The damage an attacker could cause via this vulnerability would be heavily dependent on exactly which services he could send malformed requests to. In some cases, the attacker might be able to deny particular services to legitimate users. In others, it could be possible to cause an affected system to fail altogether and require rebooting.

What would determine which services the attacker could send malformed requests to?

The most important factor would be which services are installed on the machine. For instance, if neither Exchange nor SQL Server were installed on the machine, the attacker clearly would not be able to exploit any of the vulnerabilities in those services.

Could a firewall prevent an Internet-based attacker from exploiting the vulnerability?

Yes. If the port on which an affected RPC server listens were blocked, an Internet-based attacker would not be able to deliver requests to the server, and would therefore be unable to exploit the vulnerability.

What does the patch do?

The patch eliminates the vulnerability by introducing proper validation checking into the affected RPC servers.

### ADDITIONAL INFORMATION

The information has been provided by <mailto:[secnotif@MICROSOFT.COM](mailto:secnotif@MICROSOFT.COM)>  
Microsoft Product Security.

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

[list-unsubscribe@securiteam.com](mailto:list-unsubscribe@securiteam.com)

In order to subscribe to the mailing list, simply forward this email to: [list-subscribe@securiteam.com](mailto:list-subscribe@securiteam.com)

## Hypermail Development List: [NT] Malformed RPC Request Can Caus

=====  
=====

**DISCLAIMER:**

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.

---

- *Next message:* [support@securiteam.com](mailto:support@securiteam.com): "[\[NEWS\] Continued Threat of the "Code Red" Worm](#)"
- *Previous message:* [support@securiteam.com](mailto:support@securiteam.com): "[\[NEWS\] IBM AlphaWorks TFTP Server for Java Directory Traversal](#)"
- *Messages sorted by:* [\[ date \] \[ thread \] \[ subject \] \[ author \] \[ attachment \]](#)