

# [NT] Server Response to SMTP Client EHLO Command Results In Buffer Overrun

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-07/0113.html>

---

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

*Date:* 07/25/02

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

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

Date: Thu, 25 Jul 2002 12:17:21 +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>

-- promotion

When was the last time you checked your server's security?

How about a monthly report?

<http://www.AutomatedScanning.com> -- Know that you're safe.

-----

Server Response to SMTP Client EHLO Command Results In Buffer Overrun

---

## SUMMARY

The Internet Mail Connector (IMC) enables Microsoft Exchange Server to communicate with other mail servers via SMTP. When the IMC receives an SMTP, extended Hello (EHLO) protocol command from a connecting SMTP server, it responds by sending a status reply that starts with the following:

```
250-< Exchange server ID >Hello < Connecting server ID >
```

Where:

\* < Exchange server ID > is the fully-qualified domain name (FQDN) of the Exchange server

\* < Connecting server ID > is either the FQDN or the IP address of the server that initiated the connection. The FQDN would be used if the Exchange5.5 IMC were able to resolve this information through a reverse DNS lookup; the IP address would be used if a reverse DNS lookup was not possible or failed to resolve the connecting servers IP address.

A security vulnerability results because of an unchecked buffer in the IMC code that generates the response to the EHLO protocol command. If the total length of the message exceeds a particular value, the data would overrun the buffer. If the buffer were overrun with random data, it would

## Securiteam: [NT] Server Response to SMTP Client EHLO Command Results In Buffer Overrun

result in the failure of the IMC. If, however, the buffer were overrun with carefully chosen data, it could be possible for the attacker to run code in the security context of the IMC, which runs as Exchange5.5 Service Account.

It is important to note that the attacker could not simply send data to the IMC in order to overrun the buffer. Instead, the attacker would need to create a set of conditions that would cause the IMC to overrun its own buffer when it generated the EHLO response. Specifically, the attacker would need to ensure that a reverse DNS lookup would not only succeed, but would provide an FQDN whose length was sufficient to result in the buffer overrun.

### DETAILS

#### Affected Software:

- \* Microsoft Exchange Server 5.5

#### Technical Details:

IMC is Microsoft's implementation of SMTP, which is used to facilitate the majority of email transactions on the Internet. SMTP consists of several basic operations that email clients and servers use to identify one another and deliver email. The "EHLO" command is one of these basic operations. A flaw exists in how the Exchange IMC handles EHLO commands, which are used to query other servers to obtain a list of supported SMTP operations. When an EHLO command is executed, the queried server attempts to identify the client by way of a reverse DNS lookup.

When an email client connects to the SMTP service and issues an EHLO command, the server formulates the following response to be delivered to the client:

```
[email server name] hello [client DNS name]
```

The [email server name] is the name of the system running the email server. The [client DNS name] is the name the IMC obtains by performing a reverse DNS name lookup on the client IP address.

Although DNS names can be up to 255 characters in length, the stack buffer used to formulate the message is not large enough to accommodate the entire message. Specifically, a flaw exists in that the buffer is too small for the email server name, " hello " text, and the client DNS name. Therefore, with a valid DNS reverse lookup address, an attacker can trigger the buffer overflow vulnerability. Since a test EHLO command can be issued to query the email server name, the buffer overflow can be triggered very reliably. The IMC service runs under the superuser, or SYSTEM security context. This vulnerability can be exploited by attackers using their own DNS server and controlling reverse lookup responses, or by employing DNS spoofing techniques.

Mitigating factors:

\* Creating an environment in which the IMC's reverse DNS lookup would not only succeed but also result in the buffer being overrun would be difficult. The attacker could set up a rogue DNS server and manually populate the bogus FQDN information on it, but in this would require that the attacker have some means of forcing the IMC to consult the rogue DNS server when performing the reverse DNS lookup.

\* The IMC can be disabled for cases where SMTP support is not needed. If this has been done, the vulnerability could not be exploited.

\* Customers can disable Reverse DNS lookup on EHLO by setting a registry key as defined in Q190026. The vulnerability could not be exploited on a system configured in such a way.

\* If the buffer overrun caused the IMC to fail, normal service could be restored by restarting the Exchange 5.5 IMC service.

Patch availability:

Download locations for this patch

\* Microsoft Exchange 5.5 Service Pack 4:

<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=40666>  
<http://www.microsoft.com/Downloads/Release.asp?ReleaseID=40666>

What's the scope of the vulnerability?

This is a buffer–overrun vulnerability. In the worst case, it could enable an attacker to gain complete control over an affected Exchange server. In the more likely case, it could enable an attacker to cause the Exchange service to fail.

An attacker seeking to exploit this vulnerability would face a very daunting challenge. In most cases, exploiting the vulnerability would require the attacker to not only be able to establish a connection with a vulnerable mail server, but also be able to force the server to consult a DNS server of the attacker's choice during the attack.

What causes the vulnerability?

There is an unchecked buffer in the code within the Exchange 5.5 Internet Mail Connector that responds to the Extended Hello command. Under the conditions discussed below, it could be possible to cause the buffer to be overrun.

What is the Internet Mail Connector?

The Internet Mail Connector (IMC) is a component in Exchange Server that enables Exchange to interoperate with other vendors' mail servers. Exchange can use either of two protocols to communicate with other mail servers: a proprietary protocol that is used when communicating with another Exchange server; and Simple Message Transfer Protocol (SMTP), an industry–standard protocol that is used when communicating with third–party servers. The IMC is the part of Exchange that handles communications via SMTP. The Exchange IMC does not install by default. It is also sometimes referred to as the Exchange Server Internet Mail Service.

What's the Extended Hello command?

To explain the Extended Hello (EHLO) command, we first need to discuss SMTP Service Extensions. Although the SMTP protocol specifies a full set of basic mail operations, there also is a set of additional services and functions, called SMTP extensions that many mail servers support. These are specified in RFC 1869. When two SMTP-based mail servers start a conversation, it's important for the server initiating the connection to learn which, if any, of the SMTP Service Extensions are supported by the other mail server. The EHLO command is part of this process. The message exchange between the two servers proceeds as follows:

- 1) Server 1 establishes a connection to Server 2.
- 2) Server 2 sends a "banner".
- 3) Server 1 sends an EHLO, so that it can determine what SMTP extensions Server 2 supports.
- 4) If Server 2 supports EHLO, Server 2 responds to the EHLO by sending a "Hello", identifying itself and Server 1 as the two participants in the connection and a listing all of the Server Extensions it supports.
- 5) Server 1 begins levying requests upon Server 2.

What's wrong with how the EHLO command in the Exchange 5.5 IMC is implemented?

The IMC correctly handles the incoming EHLO command. However, the software that the IMC uses to formulate its response – specifically, the code used to construct the Hello in step 4 of the preceding question – contains an unchecked buffer. It could be possible for an attacker to construct a scenario in which responding to the Hello command would, through a fairly complex process, cause the IMC to generate data that would overrun its own buffer.

What would this enable an attacker to do?

Like many buffer overruns, this one could be used to accomplish either of two goals depending on the exact data that was provided. If the buffer were overrun using random data, the effect would be to cause the IMC to fail. On the other hand, if the buffer was overrun using carefully selected data, it could be possible to, in essence, modify the IMC process to perform tasks of the attacker's choosing. Because the IMC runs with system-level privileges, this would grant the attacker complete control over the server.

How might an attacker exploit this vulnerability?

Exploiting the vulnerability would be simple in theory: the attacker would need to create a suitable environment, and then trigger an attack by connecting to a vulnerable Exchange 5.5 IMC and sending an EHLO command. Creating a suitable environment, however, could be quite difficult.

What do you mean by "a suitable environment"?

When the IMC responds to an EHLO command, it creates an initial response that identifies both servers, as discussed above. It identifies itself using its fully qualified domain name (FQDN) (e.g., "mailserver.microsoft.com"). It identifies the other server through either

## Securiteam: [NT] Server Response to SMTP Client EHLO Command Results In Buffer Overrun

its IP address or its FQDN, depending on the circumstances.

By a "suitable environment", we mean one where the following conditions are true:

- \* The IMC could determine the other server's FQDN.
- \* The length of the IMC's own FQDN, plus that of the other server's FQDN, exceeded a particular value.

These conditions would cause the IMC to overrun its buffer, and this would cause the IMC to fail. To cause the IMC to overrun its buffer and run code of the attacker's choice, an additional condition would be required, namely, the other server's FQDN would need to include executable code.

What determines whether the IMC uses the other server's IP address or its FQDN?

The IMC always tries to use the other server's FQDN. However, to do this, it needs to do a reverse DNS lookup – that is, it needs to consult a DNS server, provide it with the IP address, and receive the corresponding FQDN in return. If the reverse DNS lookup failed for any reason, the IMC would use the IP address, and this would never cause the buffer to overrun.

How difficult would it be for the attacker to make the reverse DNS lookup succeed?

The attacker would need to provide bogus data to nearby DNS servers, and wait for the data to propagate to the DNS server being used by the vulnerable mail server. However, there is a hitch. The relevant industry standard places a maximum on how long an FQDN can be, and in most cases, this value is smaller than what is needed to overrun the buffer. Thus, standards-compliant DNS servers likely would not accept the bogus data.

Instead, it is likely that the attacker would need to set up a DNS server and manually insert the bogus data. However, that would mean that the attacker would need to ensure that the vulnerable IMC consulted the attacker's DNS server. Clearly, this would make exploiting the vulnerability quite difficult.

Is it possible to disable reverse DNS lookup?

Customers can disable Reverse DNS lookup on EHLO by setting a registry key as defined in Q190026. Customers that do this are protected from the buffer overrun.

Customers that do this are protected from the buffer overrun.

When reverse DNS lookup is disabled, the Internet Mail Service will no longer resolve the host name in the "Received From" portion of the SMTP message header to the fully qualified domain name, but instead use the Internet Protocol (IP) address in the form nnn.nnn.nnn.nnn. If the address is already in Internet Protocol (IP) form, the address will remain as such.

Securiteam: [NT] Server Response to SMTP Client EHLO Command Results In Buffer Overrun

Does the vulnerability affect Exchange 2000?

No.

Does the vulnerability affect the SMTP service that ships in Windows 2000?

No.

How does the patch eliminate the vulnerability?

The patch institutes proper buffer handling in the Exchange 5.5 IMC code.

ADDITIONAL INFORMATION

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

Microsoft Product Security and <mailto:[xforce@iss.net](mailto:xforce@iss.net)> X-Force.

=====

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)

=====

=====

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.

- 
- *Previous message:* [support@securiteam.com](mailto:support@securiteam.com): "[NEWS] Heap Overflow in Solaris cachefs Daemon"
  - *Messages sorted by:* [\[ date \]](#) [\[ thread \]](#) [\[ subject \]](#) [\[ author \]](#) [\[ attachment \]](#)