

[REVS] Writing Small Shellcode In Windows

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2005-09/0083.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 09/26/05

To: list@securiteam.com

Date: 26 Sep 2005 13:14:04 +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

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

Writing Small Shellcode In Windows

SUMMARY

This paper describes an attempt to write Win32 shellcode that is as small as possible, to perform a common task subject to reasonable constraints. The solution presented implements a bindshell in 191 bytes of null-free code, and outlines some general ideas for writing small shellcode.

DETAILS

Introduction:

Size is important for shellcode because when exploiting vulnerabilities in compiled software we are often constrained in the amount of data we can work with. Smaller solutions than ours are certainly possible, but at this size the amount of work involved increases exponentially as each additional byte is trimmed from the code.

It is assumed that the reader has some familiarity with x86 assembly language.

The task to be performed by our code is as follows:

1. Bind a shell to port 6666.
2. Allow one connection to the shell.
3. Exit cleanly.

Securiteam: [REVS] Writing Small Shellcode In Windows

It must work on Windows NT4, 2000, XP and 2003, and will be launched using:

```
void main()
{
    unsigned char sc[256] = "";
    strncpy(sc,
        "shellcode goes here",
        256);
    __asm
    {
        lea eax, sc
        push eax
        ret
    }
}
```

Hence, we can observe the following:

- * The shellcode cannot contain any null bytes.
- * The shellcode must be run from the stack.
- * Winsock has not been initialised.
- * We may assume that eax points to the start of our code.

Our full solution is in the Appendix to this paper. First, we present some notes on the approach taken and some of the details of the solution.

The full whitepaper can be downloaded from:

<<http://www.ngssoftware.com/papers/WritingSmallShellcode.pdf>>
<http://www.ngssoftware.com/papers/WritingSmallShellcode.pdf>.

ADDITIONAL INFORMATION

The information has been provided by <<mailto:daf@ngssoftware.com>> Dafydd Stuttard.

The original article can be found at:

<<http://www.ngssoftware.com/papers/WritingSmallShellcode.pdf>>
<http://www.ngssoftware.com/papers/WritingSmallShellcode.pdf>

=====

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

In order to subscribe to the mailing list, simply forward this email to: 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.