

[UNIX] Sun AnswerBook 2 Format String and Other Vulnerabilities

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2002-08/0015.html>

From: support@securiteam.com

Date: 08/05/02

From: support@securiteam.com

To: list@securiteam.com

Date: Mon, 5 Aug 2002 19:06:22 +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.

Sun AnswerBook 2 Format String and Other Vulnerabilities

SUMMARY

Sun's AnswerBook 2 utilizes a third-party web server daemon (dwhttpd) that suffers from a format string vulnerability. The vulnerability can be exploited to cause the web server process to execute arbitrary code. The web server runs as user and group 'daemon' who, under recent installations of Solaris, owns no critical files. Typically, daemon only owns all files pertaining to the AnswerBook 2 installation. This effectively limits the severity of the vulnerability to a remote unprivileged shell.

In addition, not all AnswerBook Admin scripts require authentication, allowing the attacker to perform administrative functions without an account. Among other things, it is possible to add a new admin user or view the server's error log.

The combination of these two vulnerabilities allows for a remote exploit that can determine the exact location of its payload, requiring no guessing of return addresses or NOP padding.

DETAILS

Securiteam: [UNIX] Sun AnswerBook 2 Format String and Other Vulnerabilities

Vulnerable systems:

Solaris ab2 1.4.2 / dwhttpd 4.1a6 with patch 110011-02 (and before)

Example:

The following command will verify the vulnerability of the locally running ab2 server (requires perl and netcat):

```
% perl -e 'print"GET /";print"%x"x20;print" HTTP/1.0\r\n\r\n\r\n" | \  
nc localhost 8888
```

If a long string of hexadecimal digits appears in the error log, the server is vulnerable to the format string attack.

Fix:

The patches were released (without a Sun Security Alert or Security Bulletin) on January 31, 2001 and are available on <http://sunsolve.sun.com> <<http://sunsolve.sun.com>>

110538-01 AnswerBook 1.4.3_x86: HTTP GET overflow allows code execution
110537-01 AnswerBook 1.4.2_x86: HTTP GET overflow allows code execution
110532-01 AnswerBook 1.4.3: HTTP GET overflow allows code execution
110531-01 AnswerBook 1.4.2: HTTP GET overflow allows code execution

The patches have also made their way into the recommended patch clusters.

The script execution vulnerability is yet to be resolved. This can be mitigated by removing the vulnerable scripts.

The AnswerBook2 web server has been End-of-Lifed and is no longer included with Solaris releases (as of Solaris 9).

Vendor status:

09/25/2000 – security-alert@sun.com notified about format string attack
11/07/2000 – T-patches released for format string vulnerability
01/31/2001 – Patches for format string bug released to SunSolve
03/14/2001 – Sun notified about script execution vulnerabilities – Format string patches worked into recommended patch clusters
08/01/2002 – Advisory released

Technical details:

Format String Vulnerability:

User-supplied input from a GET request is used in a format string in a call to vsprintf(3s). When the file isn't found by the send_file() function, an error is logged to the ab2 log files. As the file name of the requested file is printed to the error log, vsprintf() is called with the unchecked filename. Sending a long string of "%n" formats as the filename in the GET request causes the webserver to die with a segmentation fault in vsprintf(3s).

If a long string of "%x" formats are used as the filename, values from the stack are printed out to the error log:

http-8888 [23/Sep/2000:13:09:37 -0700] warning: send-file reports: The requested object "/usr/lib/ab2/data/docs/0fea19f580073656e642d66696c65207265706f7274733a2054686520726571756573746564206f626a65637420222f7573722f6c69622f6162322f646174612f646f63732f" could not be opened!

Unauthorized Script Execution:

In DynaWeb requests, the string following the @ refers to a dwScript that generates the output. In most cases, these will be things like Ab2Admin, Ab2TocView, etc. However, browsing through the .template files in /usr/lib/ab2/dweb/data/config, we find several other interesting scripts that we can call. In particular, in ab2_admin.template, we find AdminViewError particularly interesting for our present purposes. For example, the following URL will display the error log of the local AnswerBook2 server:

<http://localhost:8888/ab2/@AdminViewError>

There are many more (possibly more useful) scripts that we can abuse, including AdminAddadmin (add user 'foo' with password 'bar'):

<http://localhost:8888/ab2/@AdminAddadmin?uid=foopassword=bar>

Exploitability:

Because input is already bounds checked, a simple buffer overflow is not possible. However, because of the interpretation of the format string, the string can be "inflated" by the format string interpretation to overflow internal buffers. A large field width is used to "inflate" the attack string, overflowing the destination argument of vsprintf(), placing code on the stack.

Using a carefully crafted request string, it is possible to exploit the format string bug to print a pointer to the stack into the error log. From this value, we can calculate the exact address where our shellcode will be on the stack. In addition, we are able to bypass authentication and executing scripts directly. This will allow us to retrieve the error log and parse our stack pointer from it.

Because the overflow happens after the HTTP request is parsed, there can be no space (0x20) or '?' (0x3f) characters in the shellcode, frame pointer or return address. Devising shellcode encoded without these bytes is relatively simple and space bytes in the frame pointer or return address (quite common under some Solaris revisions) can be encoded by creative use of the format string interpretation.

ADDITIONAL INFORMATION

The information has been provided by <<mailto:gandi@mindless.com>> gandi.

=====

Securiteam: [UNIX] Sun AnswerBook 2 Format String and Other Vulnerabilities

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.

-
- *Previous message:* support@securiteam.com: "[NT] LCC-Win32 Information Leakage"
 - *Messages sorted by:* [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#) [\[attachment \]](#)