Case Study: Code Red

Author: Jedidiah R. Crandall, crandaj@erau.edu

This Document was Funded by the National Science Foundation
Federal Cyber Service Scholarship For Service Program:
Grant No. 0113627

Distributed July 2002

Embry-Riddle Aeronautical University • Prescott, Arizona • USA

Case Study: Code Red

- What happened?
- How did it happen?
- Why did it happen?
- When did it happen?
- What does this mean?

What happened?

- The Code Red worm exploited a buffer overflow in Microsoft's IIS server, defaced web sites on English-language servers, and made a failed attempt at a denial-of-service attack on www.whitehouse.gov.
- The Code Red II worm exploited the very same vulnerability, except it installed a back door designed to make your entire hard drive available to attackers over the Internet.
- Between the two worms, about 800,000 machines infected and an estimated $2.5 billion in damages, lost productivity, and clean-up costs.

How did it happen?

A server running Microsoft's IIS will send you a web page if you make a request to that server by telling it what you want (for example, you might tell www.momscookies.com that you want the hypertext file /oatmeal/raisin.html by typing http://www.momscookies.com/oatmeal/raisin.html).

The string you send is stored in one buffer, which does not overflow because it was properly bounds-checked. Each character is an ASCII character which takes one byte to store.

If you requested some other http service, though, this buffer might be reformatted into UNICODE (used for international character sets, 1 character = 2 bytes) and stored in another buffer.

It was this other buffer that overflowed because there was no bounds checking to make sure the UNICODE buffer was twice as big as the ASCII buffer.

While it is not easy to exploit this kind of buffer overflow, it proved to not be impossible. The buffer overflow allowed the attack code, which was included in the request string, to be executed.

When did it happen?

18 June 2001- eEye Digital security reports the vulnerability

18 June 2001- Microsoft releases a patch

19 June 2001 – CERT Advisory CA-2001-13 released

12 July 2001 – First incarnation of Code Red released, doesn't spread as well as it could

19 July 2001 – Second incarnation of Code Red released, nearly the same code but it spreads much better, failed attempt at a denial-of-service attack on www.whitehouse.gov (100's of thousands of machines infected)

19 July 2001 – CERT advisory CA-2001-19 released

31 July 2001 – CAIDA follow-up survey shows that nearly a third of the machines infected by Code Red were still not patched

4 August 2001 – 16 days later, Code Red II is released, exploiting the very same vulnerability, but installing a back door on infected machines. 100's of thousands more machines are infected or re-infected. Code Red II was probably released by a different party as it shared no code with the original Code Red.

What does this mean?

Vulnerable systems had plenty of opportunity to be patched and weren't

- Systems administrators didn't know there was a vulnerability?
- Systems administrators didn't know there was a patch?
- Media didn't explain that there was a vulnerability or a patch?"

"Cyber terrorism" is a real threat

- One of the first things the worm did when it infected a machine was to check to see if that machine used English as the default language (If so the web page was defaced).
- Probably more than 200,000 of the infected machines were in the U.S., U.K., Canada, Australia, and New Zealand.
- The worm could have easily caused billions of dollars more in data loss for English-speaking-countries only.
- The denial-of-service attack on the White House would probably have succeeded had the attacker used the URL instead of the IP address.

About this Project

This presentation is part of a larger package of materials on buffer overflow vulnerabilities, defenses, and software practices.
For more information, go to: http://nsfsecurity.pr.erau.edu

Also available are:

- Demonstrations of how buffer overflows occur (Java applets)
- PowerPoint lecture-style presentations on an introduction to buffer overflows, preventing buffer overflows (for C programmers), and a case study of Code Red
- Checklists and Points to Remember for C Programmers
- An interactive module and quiz set with alternative paths for journalists/analysts and IT managers as well as programmers and testers
- A scavenger hunt on implications of the buffer overflow vulnerability

Please complete a feedback form at http://nsfsecurity.pr.erau.edu/feedback.html to tell us how you used this material and to offer suggestions for improvements.