UNIX Security Holes, Fall 2004

Course number: MCS 494, 12363
Instructor: D. J. Bernstein
Lectures: 118 DH, 11:00–11:50 MWF
Office: 410 SEO, 14:00–16:50 M

Textbook: “Exploiting software,”
by Hoglund and McGraw.
40 copies at bookstore;
less expensive online.

Prerequisite: C fluency.
Not required to know UNIX,
machine language, etc.

Goal: Understand how attackers can gain unauthorized access to today's UNIX computers.

"UNIX" is a generic term for several similar operating systems: Linux, FreeBSD, Solaris, AIX, etc.

Tangent: How *do* attackers break into UNIX computers?

Research area: How can we build a secure UNIX system?

(Vastly more difficult: How can we build a secure Windows system? Attackers have many more ways to break into Windows computers.)

What this course is not about

Attacker can break into a computer
through hardware security holes,
human security holes, etc.

Example: Attacker steals computer,
uses screwdriver to open computer,
reads credit-card number
from hard drive.

Fancier example: Attacker steals computer,
uses screwdriver to open computer,
reads credit-card number
from residual DRAM charge.

What this course is about

Attacker starts with “remote access”:
can run programs of his choice on
other computers around the Internet.

Attacker may also have “local access”:
attacker is an authorized user
of this computer; can log in
and run programs of his choice.

Some software bugs on this computer
allow attacker to
read my files, modify my files, etc.

These are **software security holes**.
Want to understand these bugs.

Robert Lemos, News.com, August 5, 2004:

“Image flaw pierces PC security

“Six vulnerabilities in a common code that handles an open-source image format could allow intruders to compromise computers running Linux and may allow attacks against Windows PCs as well as Macs running OS X.

“The security issues appear in a library supporting the portable network graphics (PNG) format, used widely by programs such as the Mozilla and Opera browsers and various e-mail clients. The most critical issue, a memory problem known as a buffer overflow, could allow specially created PNG graphics to execute a malicious program when the application loads the image.

“Among the programs that use libPNG and are likely to be affected by the flaws are the Mail application on Apple Computer’s Mac OS X, the Opera and Internet Explorer browsers on Windows, and the Mozilla and Netscape browsers on Solaris, according to independent security researcher Chris Evans, who discovered the issues.”

What are these software bugs?
How does an attacker exploit them?
We’ll see the answers.

What you have to do

Exams are 40% of your grade.

Also three types of homework.

1. Read assigned parts of textbook.
Assignment due 2004.08.25:
foreword and preface of textbook.

2. Read assigned C program excerpts
before we discuss them in class.

3. 60% of your grade:
discover 10 new security holes
in deployed UNIX software.
40 students = 400 new holes.

Collaboration is encouraged.

4 students who find 1 bug
each receive 1/4 credit for it.

Optional reading: Bugtraq,
`www.securityfocus.com/archive/1`.
Index: `www.securityfocus.com/bid`.
More than 10000 known “vulnerabilities,”
including security holes and
denial-of-service attacks; mostly accurate.

Some examples of bugs

Sendmail is a program that
accepts mail from local users,
accepts mail from the network,
delivers mail to local users,
delivers mail to the network.

1996.09.17 version: 14207 semicolons.
1999.02.04 version: 18085 semicolons.
2000.07.19 version: 26466 semicolons.
2001.09.08 version: 35171 semicolons.
2004.07.30 version: 38014 semicolons.

Sendmail's change log
reports a huge number of bug fixes,
including 58 “SECURITY” bug fixes.

What are some of the “SECURITY” bugs?