# Basic Exploit Development – Stack Based Overflows

***Course Description:***

Ever wonder how a hacker can use a relatively small piece of code to break into a computer system? Or how Google Project Zero found the latest Windows vulnerability? Exploit development is the process of understanding how software bugs can become application vulnerabilities, and how to write code to exploit these vulnerabilities. This quick two-day crash course in high-level exploit development will hopefully not only pique the interest of how exploits work under the hood, but also help you have a better understanding of why patching is so very important.

This intermediate level technical course is targeted at penetration testers who want to take their skills beyond using someone else's tools but is recommended for anyone who wants to learn about how to take a software crash to full system shell.

***Course Overview:***

The goal and objective of this course is to show students:

- How to use random arbitrary inputs to crash an application (fuzzing)
- How to read the crash and its effect on the application with a debugger (and a little assembly language)
- Using some basic Python knowledge, how to start creating exploits for simple buffer overflows
- How to make the exploit control execution flow in the application, and eventually gain a full system shell

***Requirements:***

- PLEASE have the following installed and running before class
- A laptop running virtual machine software (VMware, VirtualBox)
- Recommended:
    - VM - Windows 7 – ASLR and DEP turned off, and network connection in "Host Only" mode
    - (VMs of Windows for testing/development purposes can be found at https://developer.microsoft.com/en-us/microsoft-edge/tools/vms/)
    - VM – Kali Linux (or your favorite version of *nix)
- Software needed:
    - Windows 7 – Immunity Debugger, Python 2.7
    - Linux – SPIKE (fuzzer), Python 2.7, Metasploit (Installed by default on Kali)
- Basic Python knowledge, including building TCP sockets
- Vulnerable software will be provided via USB Flash Drive day of course
- An absolute, overwhelming curiosity for how to break things

**True Digital Security, Inc.**

**Corporate Address**
P.O. Box 35623
Tulsa, OK 74153

**Florida Office**
1401 Forum Way
Suite 100
West Palm Beach, FL 33401

p. 800.757.6937
f. 561.835.0065

**Oklahoma Office**
1350 S. Boulder Ave
Suite 1100
Tulsa, OK 74119

p. 866.430.2595
f. 877.720.4030

**New York Office**
111 Smithtown By-pass
Suite 104
Hauppauge, NY 11788

p. 631.366.5155
f. 631.366.0979

truedigitalsecurity.com

P.O. Box 35623 Tulsa, Oklahoma 74153 | tel: (866) 430-2595 | fax: (877) 720-4030

***Day 1 –***

- Basics of Exploit Development
- Intro to Assembly Code
- Lunch
- Buffer Overflows
- Fuzzing

***Day 2 –***

- Fuzzing (con't)
- Controlling EIP
- High level overview of Shellcode
- Lunch
- Finalizing the Exploit
- CTF – Vulnserver**

** Time permitting

***Wanna Prepare More?***

- Phrack Magazine - Smashing The Stack For Fun And Profit – http://phrack.org/issues/49/14.html
- Corelan Team – Exploit Writing Tutorial 1: Stack Based Overflows – https://www.corelan.be/index.php/2009/07/19/exploit-writing-tutorial-part-1-stack-based-overflows/
- SecurityTube – Assembly Primer for Hackers – http://www.securitytube.net/video/208
- Book – Hacking: The Art of Exploitation (2nd Ed) – https://www.amazon.com/Hacking-Art-Exploitation-Jon-Erickson/dp/1593271441
- Cybrary - Exploit Development Introduction – https://www.cybrary.it/video/exploit-development-introduction-part-1/
- Bl0ckbuster (hey, that's me!) – Quick and Dirty Python Exploit Development – http://bl0ckbuster.blogspot.com/2015/03/quick-and-dirty-python-exploit.html