HOW TO UNSKID YOURSELF 101

[1.] Hacking: The Art of Exploitation, 2nd Edition

This book covers coding (C, x86 assembly), exploitation (stack overflow, heap overflow, Format String), Networking (and network-based attacks), writing shellcode, countermeasures and some crypto. It's the very first book to read since it doesn't expect you to know anything before you start, though some experience with a programming language will certainly make things a lot easier.

[2.] Web application Hacker's Handbook, 2nd Edition

Covers pretty much all areas of web application security, could be seen as a reference guide, or a book to be read from start to finish. I'd recommend reading at least the first chapters before jumping back and forth in the book.

[3.] Introductory Intel x86: Architecture, Assembly, Applications, & Alliteration

http://opensecuritytraining.info/IntroX86.html
https://www.youtube.com/watch?v=H4Z0S9ZbC0g

A video course teaching you Intel x86, something you'll really want to know if you plan on pwning gibsons. It's a long course, but absolutely amazing which gives you a real good foundation for learning Software Exploitation and Reverse Engineering. Some of the stuff covered here are also in Hacking: The Art of Exploitation, but practice makes perfect.

[4.] Exploits 1: Introduction to Software Exploits

http://opensecuritytraining.info/Exploits1.html
https://www.youtube.com/watch?v=dGyWvGmBYVw&list=PL96AB65DFCE02EE3E

Another great video course from the guys over at OpenSecurityTraining. This time we'll delve deep into the art of exploitation. The Course covers Stack Overflow, Heap Overflow, writing shellcode and an intro to exploit mitigations (DEP/NX, ASLR).

[5.] Offensive Computer Security

http://www.cs.fsu.edu/~redwood/OffensiveComputerSecurity/lectures.html

Another course, this time from FSU.
Covers:

Secure Coding in C / Code Auditing
Reverse Engineering
Fuzzing
Exploit Development

- Stack/Heap/Format String
- ret2libc
- ASLR, NX/DEP, Stack Cookies, EMET
- Return Oriented Programming (ROP)

Networking
Web application Hacking/Security

- WAF
- IDS
- SSL

Metasploit
Post Exploitation
Forenscics and Incident Response
Physical Security and Social Engineering

[6.] TrailOfBits – CTF Field Guide

https://trailofbits.github.io/ctf/index.html

A text and video course covering:

- Vulnerability Discovery

- Auditing Source
- Auditing Binaries
- Auditing Webapps

- Exploit Creation

- Binary Exploits
- Webapp Exploits

- Forensics

[7.] The Shellcoder's Handbook: Discovering and Exploiting Security Holes

Amazing book. Covers stack overflows, heap overflows, format strings, writing shellcode (duh), linux x86, Windows, Solaris, OS X, Cisco IOS, exploit mitigations, fuzzing, source code auditing, binary auditing (reverse engineering), kernel exploitation on Unix/Windows and a lot more...

Note: Not a beginners book.

[8.] A Guide to Kernel Exploitation: Attacking the Core

You wanna write kernel exploits? Of course you do. Look no further.

Note: Also not a beginners book.

Books

The following books didn't really fit in the introduction list of learning material, but are just as important if you wish to continue your security journey.

[Exploit Development]

- Gray Hat Hacking: The Ethical Hacker's Handbook, Fourth Edition
- The Mac Hacker's Handbook

[Reverse Enginnering]

- Reverse Engineering for Beginners (http://beginners.re/)
- Reversing: Secrets of Reverse Engineering
- Practical Reverse Engineering: x86, x64, ARM, Windows Kernel, Reversing Tools, and Obfuscation
- IDA Pro Book, 2nd Edition
- Hacking the Xbox: An Introduction to reverse Engineering (http://bunniefoo.com/nostarch/HackingTheXbox_Free.pdf)

[Programming]

- The C Programming Language (by K&R)
- Learn C The Hard Way (http://c.learncodethehardway.org/book/)
- Violent Python: A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers
- Gray Hat Python: Python Programming for Hackers and Reverse Engineers
- PC Assembly Language (http://www.drpaulcarter.com/pcasm/) (x86 NASM)
- Programming from the Ground Up (http://savannah.nongnu.org/projects/pgubook/) (x86 AT&T)
- Assembly Language Step by Step: Programming with Linux 3rd Edition
- 64 Bit Intel Assembly Language Programming for Linux

[Auditing and Vulnerability discovery]

- A Bug Hunter's Diary: A Guided Tour Through the Wilds of Software Security
- The Art of Software Security Assessment: Identifying and Preventing Software Vulnerabilities
- Fuzzing: Brute Force Vulnerability Discovery

[Penetration Testing]

- Metasploit Unleashed (http://www.offensive-security.com/metasploit-unleashed/Main_Page)
- Metasploit: The Penetration Tester's Guide
- The Hacker Playbook: Practical Guide To Penetration Testing
- RTFM: Red Team Field Manual

[Web Security]

- The Tangled Web: A Guide to Securing Modern Web Applications
- SQL Injection Attacks and Defense, Second Edition
- The Browser Hacker's Handbook
- Web Application Obfuscation: '-/WAFs..Evasion..Filters//alert(/Obfuscation/)-'

[Malware, Forensics and Anti-Forensics]

- The Rootkit Arsenal: Escape and Evasion: Escape and Evasion in the Dark Corners of the System
- Designing BSD Rootkits: An Introduction to Kernel Hacking
- Rootkits: Subverting the Windows Kernel
- Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software

[Mobile Security]

- iOS Hacker's Handbook
- Android Hacker's Handbook
- The Mobile Application Hacker's Handbook

Video Courses

Exploits 1: Introduction To Software Exploits

http://opensecuritytraining.info/Exploits1.html
https://www.youtube.com/playlist?list=PL96AB65DFCE02EE3E

Exploits 2: Exploitation in the Windows Environment

http://opensecuritytraining.info/Exploits2.html
https://www.youtube.com/playlist?list=PL9F9E52502327B1CA

Introduction To Reverse Engineering Software

http://opensecuritytraining.info/IntroductionToReverseEngineering.html
https://www.youtube.com/playlist?list=PLUFkSN0XLZ-nXcDG89jS9iqKBnNHmz7Qw

Intermediate Intel x86: Architecture, Assembly, Applications, & Alliteration

http://opensecuritytraining.info/IntermediateX86.html
https://www.youtube.com/playlist?list=PL8F8D45D6C1FFD177

MIT 6.858: Computer Systems Security

http://css.csail.mit.edu/6.858/2014/
https://www.youtube.com/watch?v=M2gc6b1hmk8&index=1&list=PLA6Ht2dJt3SLQmKhygx8HfwV_hxuPPCea

Tutorial Series

Exploit Development:
https://www.corelan.be/index.php/articles/
http://expdev-kiuhnm.rhcloud.com/
http://www.securitysift.com/windows-exploit-development-part-1-basics/
http://www.fuzzysecurity.com/tutorials.html

Tools

[Fuzzing]

american fuzzy lop

- "American fuzzy lop is a security-oriented fuzzer that employs a novel type of compile-time instrumentation and genetic algorithms to automatically discover clean, interesting test cases that trigger new internal states in the targeted binary. This substantially improves the functional coverage for the fuzzed code."

http://lcamtuf.coredump.cx/afl/

PEACH Community Fuzzer

- "Peach is a SmartFuzzer that is capable of performing both generation and mutation based fuzzing."

http://community.peachfuzzer.com/

SPIKE

- "Written in C, exposes a custom API for fuzzer development. Probably the most widely used and popular framework."

http://www.fuzzing.org/ or http://www.immunitysec.com/resources-freesoftware.shtml

Zulu – The Interactive Fuzzer

- "Zulu is an interactive GUI-based fuzzer. It is as much as possible, input and output-agnostic so once you are happy with using the fuzzing engine that's driven by the GUI you are only limited by the input and output modules that have been developed for it.“

https://github.com/nccgroup/Zulu

sulley

- “A pure-python fully automated and unattended fuzzing framework.”

https://github.com/OpenRCE/sulley

zzuf - multi-purpose fuzzer

- “zzuf is a transparent application input fuzzer.”

http://caca.zoy.org/wiki/zzuf

More information on fuzzers:
http://www.fuzzing.org/wp-content/sample_chapter.pdf
http://www.blackhat.com/presentations/bh-usa-09/EDDINGTON/BHUSA09-Eddington-DemystFuzzers-PAPER.pdf

[Exploit Development]

mona.py

- A debugger plugin for Windows exploit development.

https://github.com/corelan/mona
https://www.corelan.be/index.php/2011/07/14/mona-py-the-manual/

PEDA

- Python Exploit Development Assistance for GDB

https://github.com/longld/peda

pwntools

- pwntools is a CTF framework and exploit development library. Written in Python, it is designed for rapid prototyping and development, and intended to make exploit writing as simple as possible.

https://pwntools.readthedocs.org/en/2.2/
https://github.com/Gallopsled/pwntools

Metasploit Framework

- This framework has some pretty great tools for exploit development. I encourage you to go read the wiki which will help you setup a development environment and getting started:

https://github.com/rapid7/metasploit-framework/wiki

[Debuggers]

(will finish this later)

GDB
radare2

WinDBG
OllyDBG
Immunity Debugger

TODO:

- add more content (whitepapers etc)
- write an introduction
- organize shit better

xoxo,
pantsu && lsd