I'm not robot

reCAPTCHA

Continue

Violent python tj o'connor pdf

This repository focuses on proposing new solutions for the exercises in violent python, a cookbook for hackers, penetration testers and security engineers by TJ. Mr. O'Connor. ISBN: 978-1-59749-957-6 A comparison between the book solution and my proposed solution will be presented. I'm doing this as a learning exercise on the following topics: Python Migration Code 2 to 3 Applying Generators Applying Functional Programming Providing a benchmark for both the solution proposed by the book and mine. I am aware that a comparison between benchmarks will not be simple, as I will add functionality to most solutions and use python3 to code my solutions instead of python2 that is used in the book. This is presented as a learning exercise, please refer to the LICENSE file in this repository for licensing and usage issues, applicable only to the code created by me. I'd like to thank T.J. O'Connor for writing the Violent Python Book, which has been an important piece in his career. By presenting this work I'm just trying to apply new concepts to an already amazing work. All the exercise solutions proposed in the book can be found here. Finally, this is the system I'm using to run my code. /:--------------:\ localhost.localdomain :-------------------:: OS: Fedora 25 TwentyFive :----------/shhOHbmp---:\ Kernel: x86_64 Linux 4.11.7-200.fc25.x86_64 /----------MOMMMNMMD ---: Uptime: 13h 18m :----------sMMMMNMNNMP. ---: Packages: - :----------:MMMdP------- ---\ Shell: - ,------------:MMMd-------- ---: Resolution: - :-----------:MMMd------- .---: FROM: GNOME :---- ----: :-- .+shhhMMMmhhy++ .------/ :- -------:MMMd------------; :- --------/MMMd------------; :- ------/hMMMy------------; :-- :d MNdhhdNMMNo------------; CPU: Intel Core i5 M 560 @ 4x 2.667GHz [52.0°C] :---:sdNMMMMNds:------------: GPU: Intel(R) Ironlake Mobile DRI Table :------::///:-------------:: RAM: 4747MiB / 5762MiB :---------------------// Marks of Recognition Lead Author of dedication - TJ O'Connor Bio Contributor Author – Rob Frost Bio Technical Editor – Mark Baggett Introduction ORGANIZATION OF THE TARGET AUDIENCE OF THE COMPANION BOOK OF SITE Chapter 1. Introduction introduction: A penetration test with python configuring your development environment The Python language Your first python programs Chapter Wrap-Up References Chapter 2. Penetration test with python introduction: would the morris worm work today? Building a port scanner building a BotNet SSH with Python Mass Compromise, connecting FTP and Web Conficker, why try hard is always good enough by writing your own proof of concept concept chapter ending Chapter Closing References Chapter 3. Forensic Investigations Python introduction: How did forensics solve btk murders where you've been?— Analysis of wireless access points in the registry using python python Retrieve deleted items in the recycle bin that investigates python application artifacts by investigating iTunes mobile backups with closing references from Chapter Python Chapter 4. Network traffic analysis with python introduction: Operation Aurora and How the Obvious was lost Where is this IP traffic directed?— Is a python response anonymous really anonymous? Analyzing LOIC traffic as H D Moore solved the Pentagon's dilemma over the rapid flow of the storm and the domain flow of Conficker Kevin Mitnick and the tcp sequence prediction by thwarting intrusion detection systems with wrap references from Chapter Scapy In Chapter 5. Wireless chaos with python introduction: Wireless Security (IN)and iceman setting up your wireless attack environment The sheep wall —passively listening to wireless secrets where your laptop has been? Python Responds Interception and Spying on UAVs with Python Detecting FireSheep Chasing with Bluetooth and Python Chapter Wrap Up References Chapter 6. Web Recon with Python Introduction: Social Engineering Today Using the Mechanize Library to Browse the Internet Scraping Web Pages with AnonBrowser Research, Investigate, Discovery Anonymous Email Mass Social Engineering Chapter Wrap-Up References Chapter 7. Antivirus evasion with python introduction: Flame On! By circumventing antivirus programs by checking for avoidance wrap up index references The rapid flow of the storm and the conficker domain flow in 2007, security researchers have identified a new technique used by the infamous Botnet Storm (Higgins, 2007). The technique, called a fast flow, used domain name service (DNS) records to hide the command and control servers that controlled the Storm botnet. DNS records typically translate a domain name to an IP address. When a DNS server returns a result, it also specifies the TTL that the IP address remains valid before the host is re-verified. The attackers behind the Storm botnet have changed the DNS records for the command and control server quite often. In fact, they used 2,000 redundant hosts spread across 384 providers in more than 50 countries (Lemos, 2007). Attackers exchanged IP addresses for the command and control server frequently and ensured dns results returned with a very short TTL. This rapid flow of IP addresses made it difficult to identify the command and control servers for the botnet and even more difficult to shut down the servers. Although rapid flow proved difficult in the fall of the Storm botnet, a similar technique used the following year helped infect seven million computers in more than two hundred countries (Binde et al., 2011). Conficker, the most successful computer worm to date, has spread a vulnerability in the Windows Service Message Block (SMB) protocol. Once infected, the vulnerable machines came into contact with a command and control server for additional instructions. Identification and e communication with the command and control server proved absolutely necessary for those involved in stopping the attack. However, Conficker generated different domain names every three hours, using the current date and time in UTC. For conficker's third iteration, this meant that 50,000 domains were generated every three hours. Attackers registered only a handful of these domains in actual IP addresses for the command and control servers. This made it very difficult to intercept and prevent traffic with the command and control server. Because the technique rotated domain names, the researchers called it domain flow. In the next section, we'll write some python scripts to detect rapid flow and domain flow in nature to identify attacks. — Tj O'Connor, Violent Python: A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers The book appears all here and there in online forums whenever there is a question How to start (/r/netsec for example).* The author's humor makes the book really fun to read, despite being a technical book.- Almost all topics are inspired from a real event (like Operation Aurora) or a common vulnerability/attack vector (like FTP Anonymous), thus providing a lot of real-life examples on how/why something is broken and what to do to exploit/fix it.- All scripts are sh The book appears all here and there in online forums whenever there is a question How to start (/r/netsec for example). :- The author's humor makes the book really fun to read, despite being a technical book.- Almost all topics are inspired from a real event (like Operation Aurora) or a common vector of vulnerability/attack (like FTP Anonymous), thus providing a lot of real-life examples about how/why something is broken and what to do to explore/fix it.- All scripts are short, self-contained with explanations, also very good and scalable code structures in general, from args analysis to the end.* Cons:- Python 2 is like from 200 a.C. or something. Also, many APIs and databases used in the book are outdated/unavailable/not compatible with Python 3, which is much more annoying than a few minor differences in language syntax.- It's not exactly beginner level. As the title of the book suggests: Cookbook, you should have some basic background knowledge in certain fields such as basic networking, FTP/SSH, scanning,...* I would recommend ?- If you are new to new beginner, regardless to coding or new to InfoSec, then probably not. Instead, I would suggest reading about basic networking.- If you already know some fundamentals, then yes. The text is really ingenious and informative than the code, not the desnate or something If you're already experienced, then probably not, most of the tools here are a bit simple compared to common tools (nmap, hydra,...). However, you can find some inspirations here. Here. I learned a lot about forensics, like using a huge database mapping IP to locations to see past connections. I would skip most of the code and go just to the text. ... more Get Violent Python now with O'Reilly learning online. O'Reilly members experience live online training, as well as books, videos, and digital content from more than 200 publishers. Violent Python shows how to move from a theoretical understanding of offensive computing concepts to a practical implementation. Instead of relying on another attacker's tools, this book will teach you how to forge your own weapons using the Python programming language. This book demonstrates how to write Python scripts to automate large-scale network attacks, extract metadata, and investigate forensic artifacts. It also shows how to write code to intercept and analyze network traffic using python, wireless creation and spoofing frames to attack wireless and Bluetooth devices, and how to mine data from popular social media sites and avoid modern antivirus. Demonstrates how to write Python scripts to automate large-scale network attacks, extract metadata, and investigate forensic artifactsSAtocoof code to intercept and analyze network traffic using python. Create and spoof wireless frames to attack wireless devices and BluetoothDesa-mine popular social media sites and prevent modern antivirus antivirus