IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/46ICT/2025/4624725/
Vol. 2025 (35), Article ID 4624725

A Penetration Testing Platform: Three Testing Tools*

Remigiusz RAJEWSKI and Jakub SKORA

Institute of Communication and Computer Networks, Faculty of Computing and Telecommunications, Poznan University of Technology, ul. Polanka 3, 61-131 Poznan, POLAND, e-mail:

Correspondence should be addressed to: Remigiusz RAJEWSKI, remigiusz.rajewski@put.poznan.pl

* Presented at the 46th IBIMA International Conference, 26-27 November 2025, Ronda, Spain

Abstract

The aim of this article is to take a deeper look at penetration testing, its theoretical and practical sides, by reviewing existing solutions and creating custom tools that can be used during such tests. Therefore, we present three pentesting tools tested on a dedicated platform: the Raspberry Pi 4B single-board computer. All three tools we created were primarily intended to streamline and automate the penetration testing tasks they addressed. We focus on simplicity; therefore, the interface makes entry easy for potential users and eliminates barriers to getting started. This graphical environment allows for use without knowledge of command syntax or the command line. Our solution combines several tools into a single program, making it easier for users to run different tests. The entire tool was created in the widely popular Python language, which facilitates the program's continued improvement and development.

Keywords: penetration testing, pentesting, automate tests

Introduction

Today, the world is changing at an incredibly rapid pace, especially the digital world. Security is one of its fundamental pillars. The number of threats that security teams must deal with is constantly growing and evolving. Cybercriminals seek to gain illegal access to data and develop increasingly advanced methods to breach security. They often target complex IT systems whose security is crucial in many areas of life. Security incidents can often involve the leakage of sensitive data or other negative situations. Such situations must be avoided. A good prevention practice is to perform frequent and effective penetration tests of environments, systems, and applications. These tests, as simulated attacks, can identify weak points in defenses and help eliminate them. Tests can be very extensive, but can also focus on specific aspects of the IT system being tested. A wide range of tools with a wide range of applications can be used to carry out such tests. Penetration testers often create their own scripts to accelerate testing and improve results. They utilize existing solutions and sometimes implement their own ideas. This allows them to achieve what is most important in testing: finding potential security incidents. Discovering, removing, or limiting their potential impact is the primary goal of penetration testing.

The need to create or adapt the available penetration testing tools to meet the requirements of a given test or organization is increasingly emerging. Creating our own tools or using appropriately modified existing solutions allows for better adaptation to the scenario and more effective evaluation results. In this article, we will present Python programs we have created to perform various penetration testing activities. They can be used successfully on Linux, specifically Kali Linux on a Raspberry Pi, for which they were created and optimized. Properly developing own penetration testing tools can often be a challenge, requiring some technical and programming knowledge, as well as experience.

Cite this Article as: Remigiusz RAJEWSKI and Jakub SKORA, Vol. 2025 (35) “A Penetration Testing Platform: Three Testing Tools" Communications of International Proceedings, Vol. 2025 (35), Article ID 4624725, https://doi.org/10.5171/2025.4624725

The rest of this paper is as follows. In Section 2 we discuss the known programs and tools used in penetration testing. In Section 3 we describe the platform used for the tests. In turn, the next section includes all the proposed programs (for scanning and brute force attacks, for sending the generated packages, and the penetration tester toolkit, respectively) prepared by us. The last section constitutes a summary.

Known Attacks and Tools

In this section, we describe briefly the already known state of the art in penetration testing focusing on the known types of attacks and tools.

Types of attacks on IT systems

Many attacks are carried out on IT systems every day. The ever-increasing digitalization of life creates new opportunities for attackers. These attacks can have various goals, such as disrupting the proper operation of systems or completely blocking them, but the primary goal is usually to steel the data. This data can then be used for further hacking activities, extorting money, obtaining confidential information, compromising the organization, etc. With knowledge of the specifics of a given IT system, we can expect specific types of attack. The familiarity with how they work makes it easier to defend against them by conducting penetration tests that simulate such attacks.

We can distinguished some kind of attacks like: Phishing, Malware, DoS & DDoS Attacks, Man-in-the-Middle (MitM) Attack, SQL Injection, Cross-Site Scripting (XSS), Zero-Day Exploit, Brute Force Attack, Credential Stuffing, and Insider Threats.

Tools for conducting penetration testing

To perform an effective penetration test, various specialized tools are typically used, which generally allow the identification and exploitation of weaknesses in IT systems. Here we only mention some of the most well-known and popular tools used for penetration testing, based on their functions, features, and applications.

We can distinguish some kind of tools like: nmap (2025), Recon-ng (2025), theHarvester (2025), Legion (2025), DMitry (2024), Wireshark (2025), Maltego (2025), OSRFramework (2025), Shodan (2025), and Spiderfoot (2025).

It is important to remember that the improper use of these tools can be illegal or violate network security policies. Always adhere to the appropriate regulations and ethical principles during penetration testing.

Password Attacks

Passwords are currently a very popular authentication method. They are used in conjunction with logins to gain access. Other authentication methods, such as multi-factor authentication using tokens, biometrics, or certificates, are becoming increasingly popular. However, most systems still use password authentication because it is easier for the end user and during the implementation stage. These passwords are often short, common, default, or contain a pattern, such as words. Generally speaking, these passwords are weak and easy to crack.

Special tools exist to test whether such passwords constitute a vulnerability. The most popular ones are: Hydra (2025), Ncrack (2025), Mimikatz (2025), John the Ripper (2025), Hashcat (2025), Medusa (2025), and Ophcrack (2025).

Used Platform

To test the tools, we used the Linux operating system, specifically the 64-bit Kali Linux 2023.1 distribution for the Raspberry Pi platform. It was designed specifically for penetration testing; therefore, we did not need to install some useful tools/programs. It is just a powerful toolbox that includes numerous applications that enable effective security testing. Among the strengths of this distribution are:

- integrated penetration testing tools – includes numerous tools for network scanning, web application security testing, password cracking, wireless network attack, and more;
- built-in security features – designed to protect users from attacks while enabling penetration testing;
- cross-platform support – can run on multiple hardware platforms;
- user community – has extensive support from the user community, allowing to get help and support when needed.

So, Kali Linux is an ideal solution for penetration testing professionals because it provides a wide range of functionalities and tools necessary to perform security tests effectively \cite{kali}.

Kali Linux was used on a Raspberry Pi 4B single-board computer. It is equipped with a Broadcom BCM2711 quad-core Cortex-A72 (ARM v8) 64-bit system-on-a-chip (SoC) processor running at 1.5 GHz. It also includes 8GB of LPDDR4-3200 SDRAM, which provides faster and more efficient memory management compared to previous Raspberry Pi models. Connectivity-wise, it offers support for 802.11ac, Gigabit Ethernet, and Bluetooth 5.0. It also has two USB 3.0 ports, two USB 2.0 ports, two micro-HDMI ports, a 3.5mm audio jack, and a microSD card slot for storing the operating system and data. We installed Kali Linux on such a 32GB card. The Raspberry Pi is capable of running a wide range of operating systems and is also compatible with various software and programming languages, making it a versatile tool. Advantages include:

- high performance;
- low cost compared to traditional computers;
- multitude of applications;
- community support;
- ease of use.

In general, the Raspberry Pi 4B 8GB is a powerful and versatile single-board computer offering impressive performance and connectivity options, making it an excellent choice for a wide range of projects and applications, including conducting a wide range of specialized penetration tests at various scales. Thanks to its wireless network adapter, it can be used to conduct a wide range of Wi-Fi network tests. Its low power consumption and small size allow to operate in a wide variety of locations. The ability to boot from a memory card allows for quick system configuration changes based on our needs. Combined with Kali Linux, it provides a very good platform for conducting a wide range of penetration tests.

We used the manufacturer's recommended *Raspberry Pi Imager* to install the new operating system. This is an easy-to-use program that creates bootable operating system images for the chosen Raspberry Pi platform. It allows users to choose from supported systems and automatically detects the media on which the system can be installed. The program allows for formatting, deleting partitions, and displaying available disk space. *Raspberry Pi Imager* is available for various platforms and offers the option of downloading the operating system directly from the Internet. It is an essential tool for anyone working with the Raspberry Pi (2025).

After the main system was ready to use, we have installed a few essentials on Kali Linux, including Python and xrdp. They communicate via ssh or rdp.

Prepared Programs

This section includes all the proposed programs for scanning and brute force attacks, for sending the generated packages, and the penetration tester toolkit, respectively. All of them were prepared by us from scratch. This allowed us to become familiar with how some already known tools are working, how they can be added to customize purposes, and so on.

A. Host Scanning and Password Brute Force Attack

The first program created in Python is designed to host scanning and password brute force attack. It uses the *Tkinter* library for the graphical interface. It has a consistent, clear, and intuitive graphical interface. The created tool can be seen in Figures 1-4. The program offers one of the most frequently used functions during testing: scanning available hosts and their ports. It also provides the ability to brute force logins and passwords on hosts to services found on ports. The script uses the widely known *nmap* tool for host discovery and port scanning, as well as *Hydra*, *Medusa*, and *Ncrack* for password attacks. With proper implementation, it is easy to expand the program with additional password-checking tools.

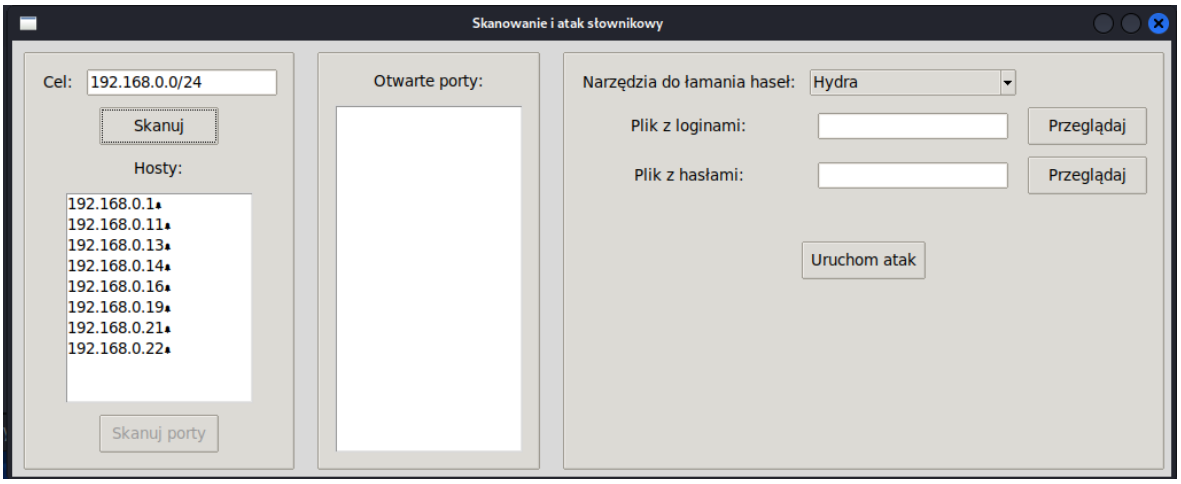

Fig 1. Host scanning and brute force attack – printscreen no 1

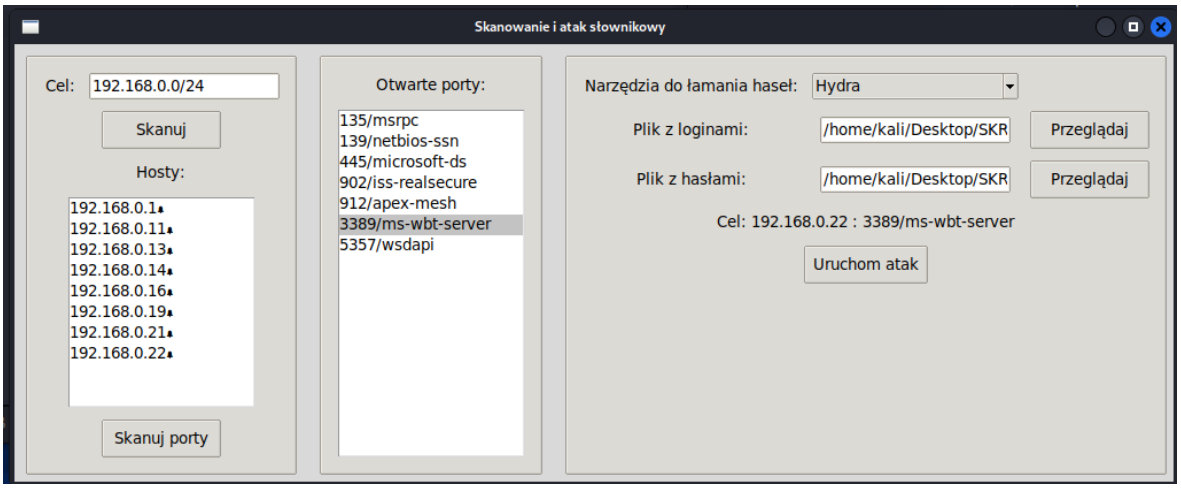

Fig 2. Host scanning and brute force attack – printscreen no 2

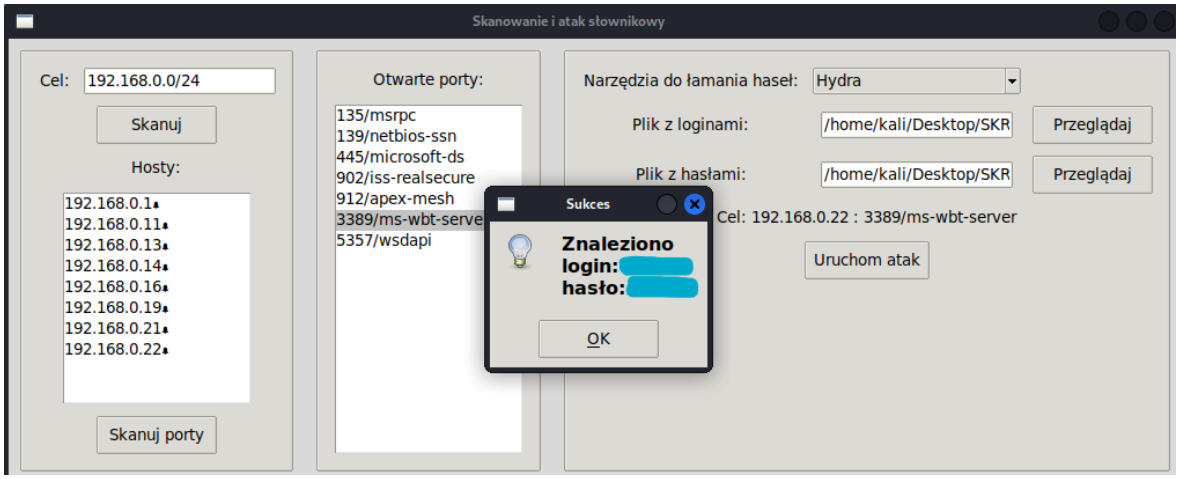

Fig 3. Host scanning and brute force attack – printscreen no 3

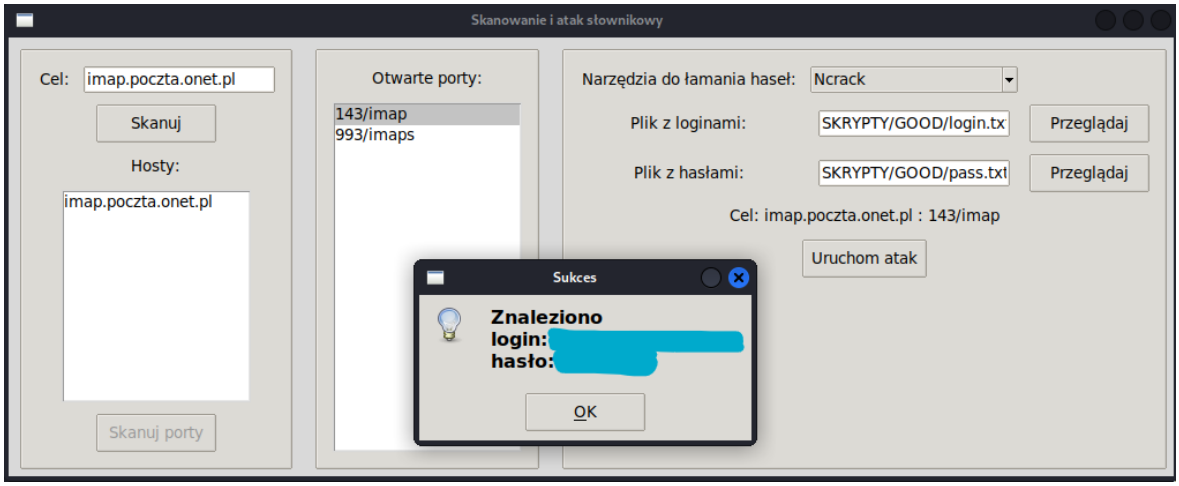

Fig 4. Host scanning and brute force attack – printscreen no 4

The program can scan subnets by providing an address and mask, and then select one from the list of discovered hosts and check for open ports. We can also enter a specific IP address or mnemonic to immediately scan available services on the ports. If the goal was to check hosts and open ports, the operation can be considered complete. If we also want to test the password, we can just continue and go with further steps as well. Once the appropriate port/service pair is selected, we can choose the tool to use for checking logins and passwords – *Hydra*, *Medusa*, or *Ncrack*. Next, we need to specify the login file and the password file to be checked. This is followed by a summary of the target, the IP address, and port of the service for which the login and password were checked. During each phase of the operation, from host scanning, through port scanning, to password checking, a window appears informing us of the ongoing operation. In addition, at the end of the password check, information about the result is displayed. If successful, the combination found is also displayed.

In summary, the tool helps scan available hosts, check their ports, and test the security of individual services, such as SSH, IMAP, or SMTP. Automation of operation and cooperation of several different tools enable more effective penetration testing. Using the program does not require knowledge of the syntax of any commands, and, thanks to the graphical interface, it does not require using the command line as well.

Sending own Generated Packages

The second program is a tool designed to perform penetration tests of IT networks. It can also be used to diagnose various network connectivity issues. This program generates custom IP version 4 packets. It also includes predefined protocols for faster and easier use. Like the previous program, it features a simple and minimalist graphical interface, maintaining a similar style to its predecessor. This second tool is shown in

Figures 5-8. It allows us to send packets that we just created by selecting or entering the required data and values for the appropriate parameters. In version 1 it supports sending packets for the ARP, IPv4, ICMP, TCP, UDP, DHCP, and DNS protocols. Due to its modular design, the tool can be easily expanded to support additional protocols in the next version. The operation of the program can be divided into three parts:

- The first involves entering the source and destination IP addresses and selecting the protocol.
- The next step depends on the selected protocol. If we choose ARP, we need to select the message type and provide both the source and destination physical addresses. Selecting IP allows us to enter values for each IPv4 header field. For ICMP, only the message type can be selected. In turn, DHCP allows us to select the message type and enter the IP addresses used by this protocol. TCP allows us to enter a port and select a flag. The procedure for other protocols is similar, depending on their specificity.
- The third step is sending. The "Send packet" button ("Wyślij pakiet" in figures) can be clicked any number of times; for example, if we want to send the packet five times, we just need to click the button five times.

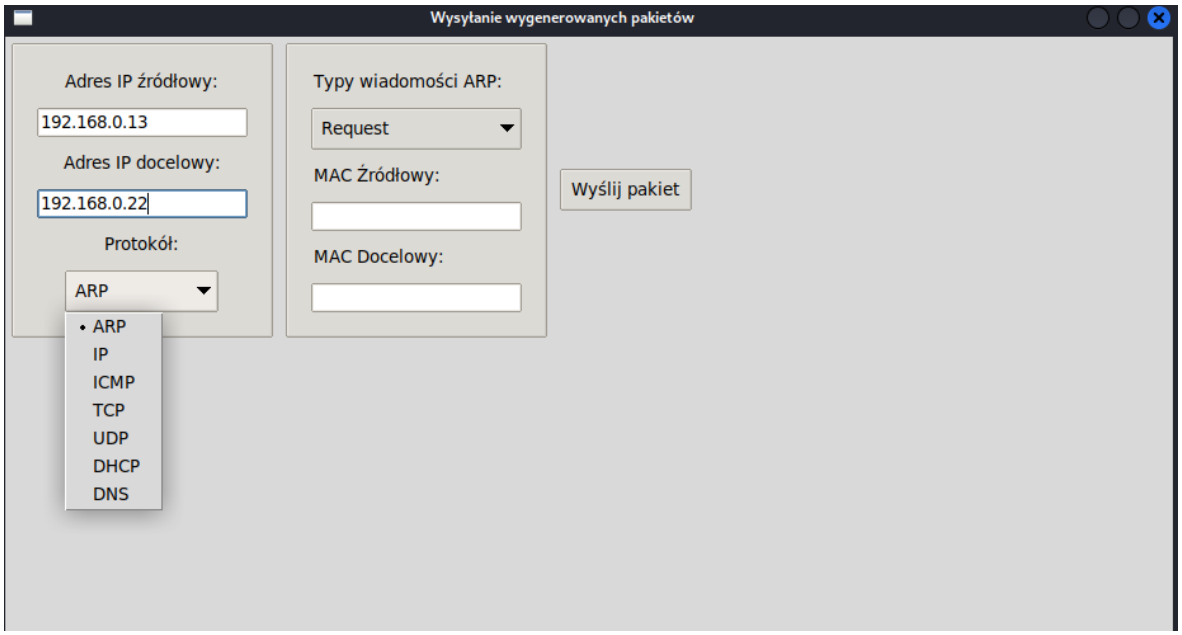

Fig 5. Sending own generated packages – printscreen no 1

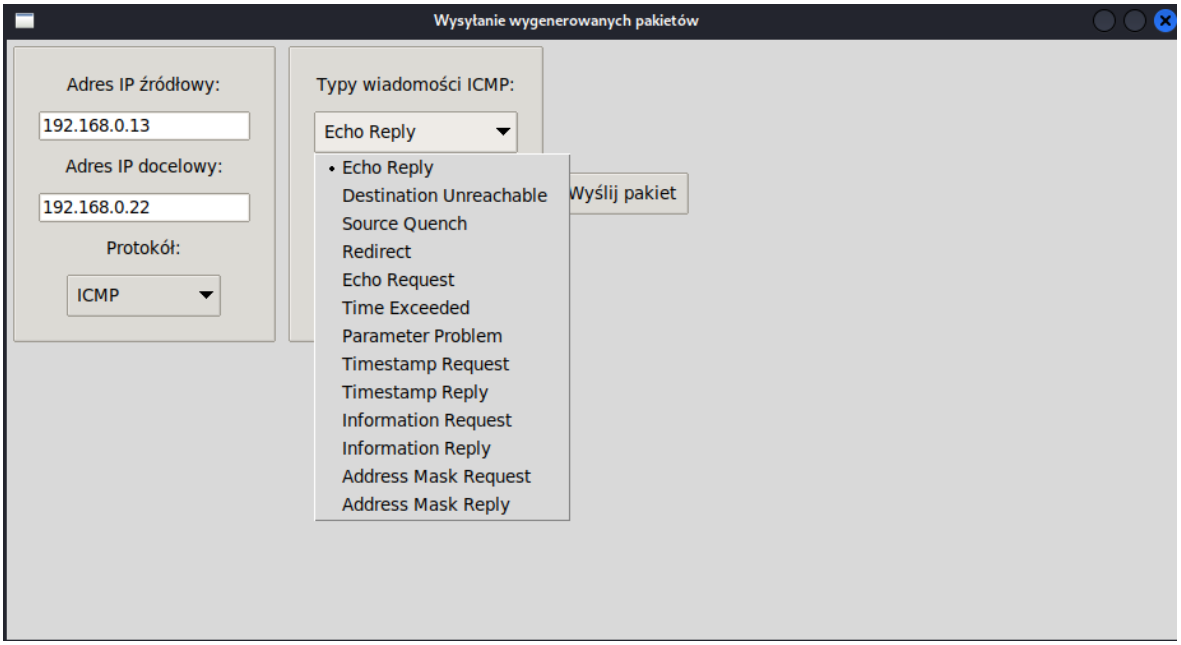

Fig 6. Sending own generated packages – printscreen no 2

In summary, this program streamlines the work of those involved with IT networks and their security. Its typical use is to check the response to a given packet, for example, on a firewall or other device or application. It also allows us to check the flow of the packet or manipulate it to produce a desired effect. It is a quick and easy-to-use tool that can be successfully applied to a wide variety of IT network-related tasks. Like the first tool (introduced in Section 3A), this one is also written in Python using many popular libraries, including *Tkinter* and *Scapy* for creating a graphical interface and sending packets, respectively.

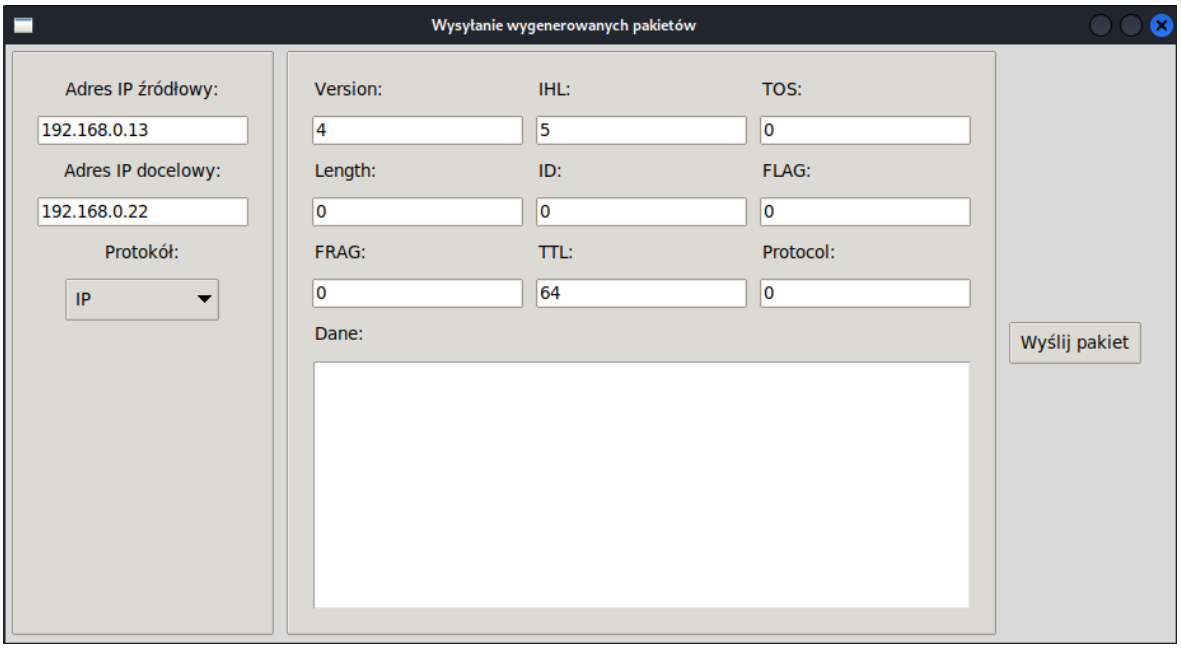

Fig 7. Sending own generated packages – printscreen no 3

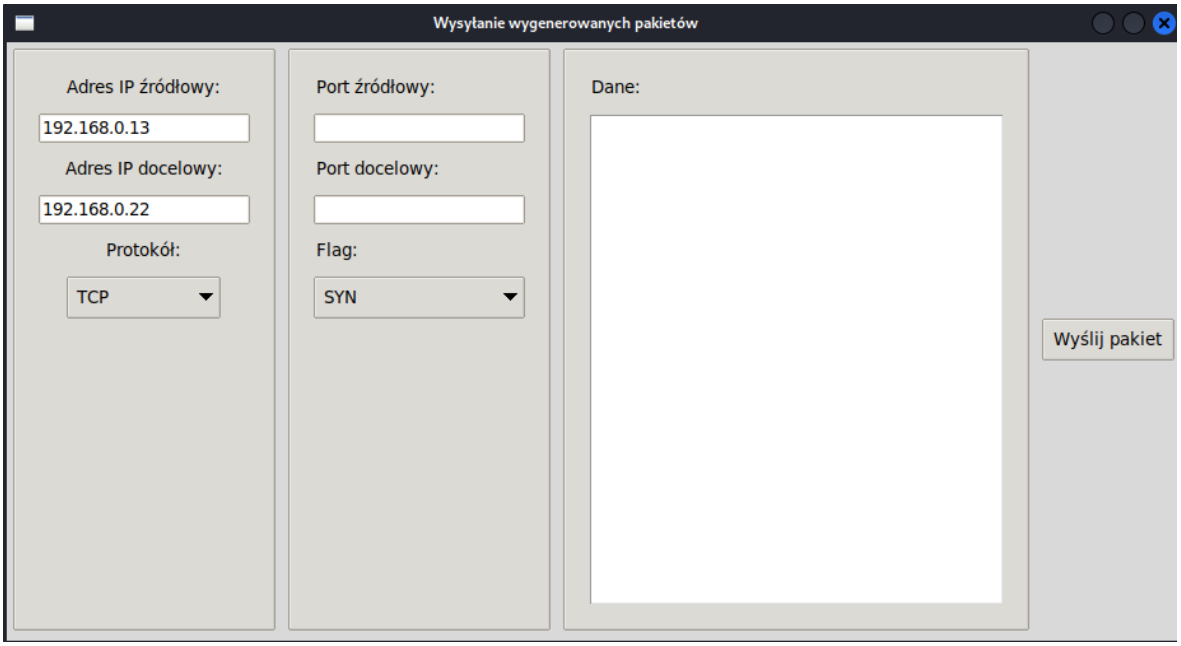

Fig 8. Sending own generated packages – printscreen no 4

Penetration Tester Toolkit

The third program is a solution that can be called a suite of tools that can be useful during penetration testing, but also in everyday work. The tool features a graphical interface that is aesthetically pleasing and responsive, similar to previous solutions. It is shown in Figures 9-12. It consists of two parts:

- The first is a menu of buttons at the top with functionalities.
- The second, depending on the function selected in the previous part, shows which tool the selected option provides.

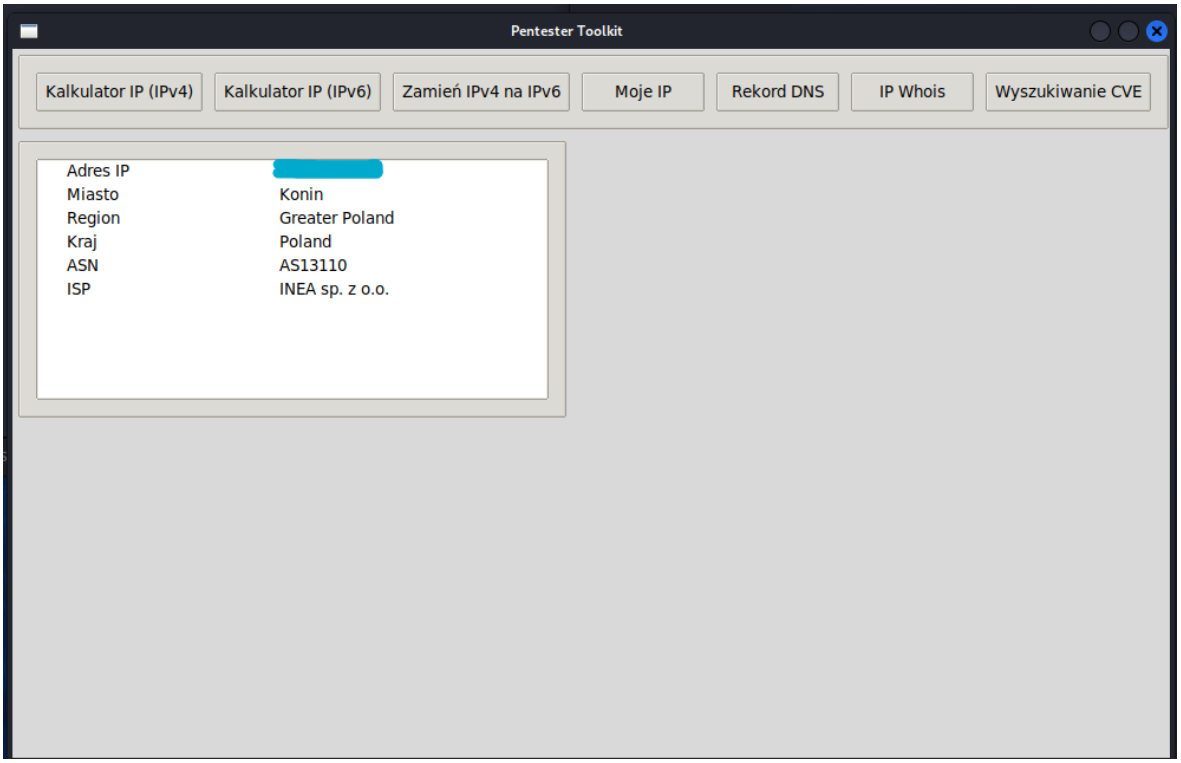

Fig 9. Penetration tester toolkit – printscreen no 1

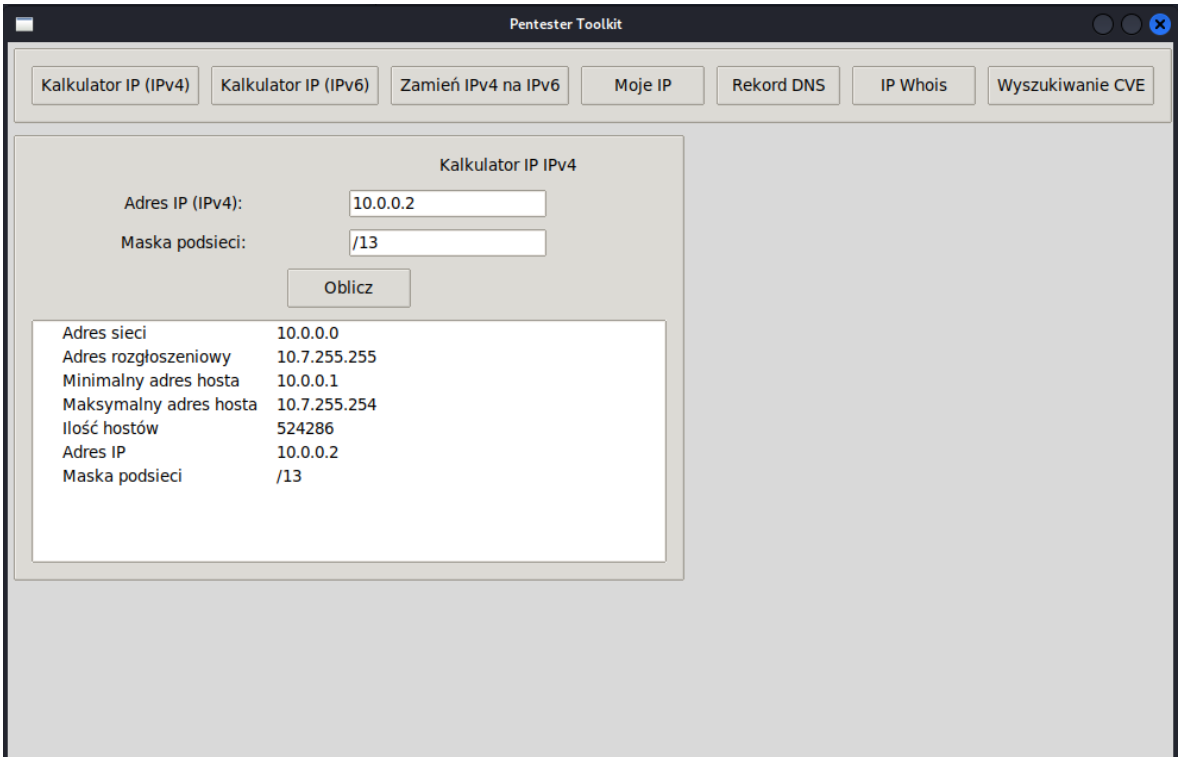

Fig 10. Penetration tester toolkit – printscreen no 2

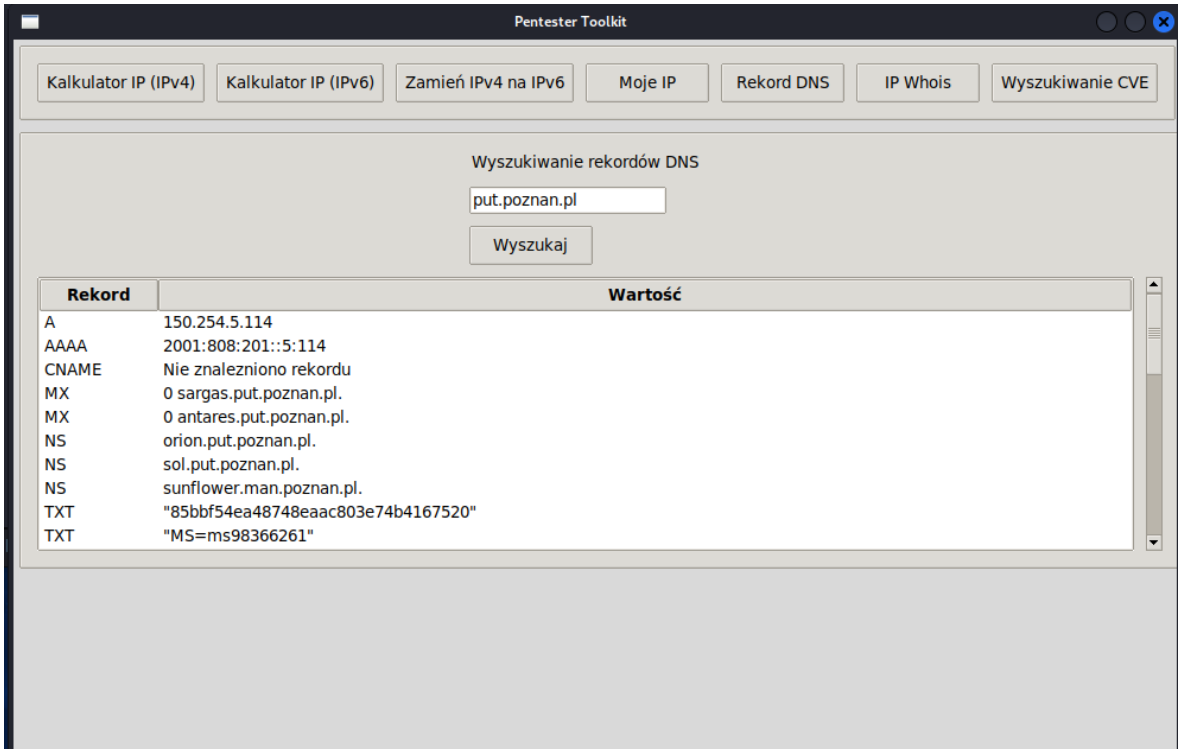

Fig 11. Penetration tester toolkit – printscreen no 3

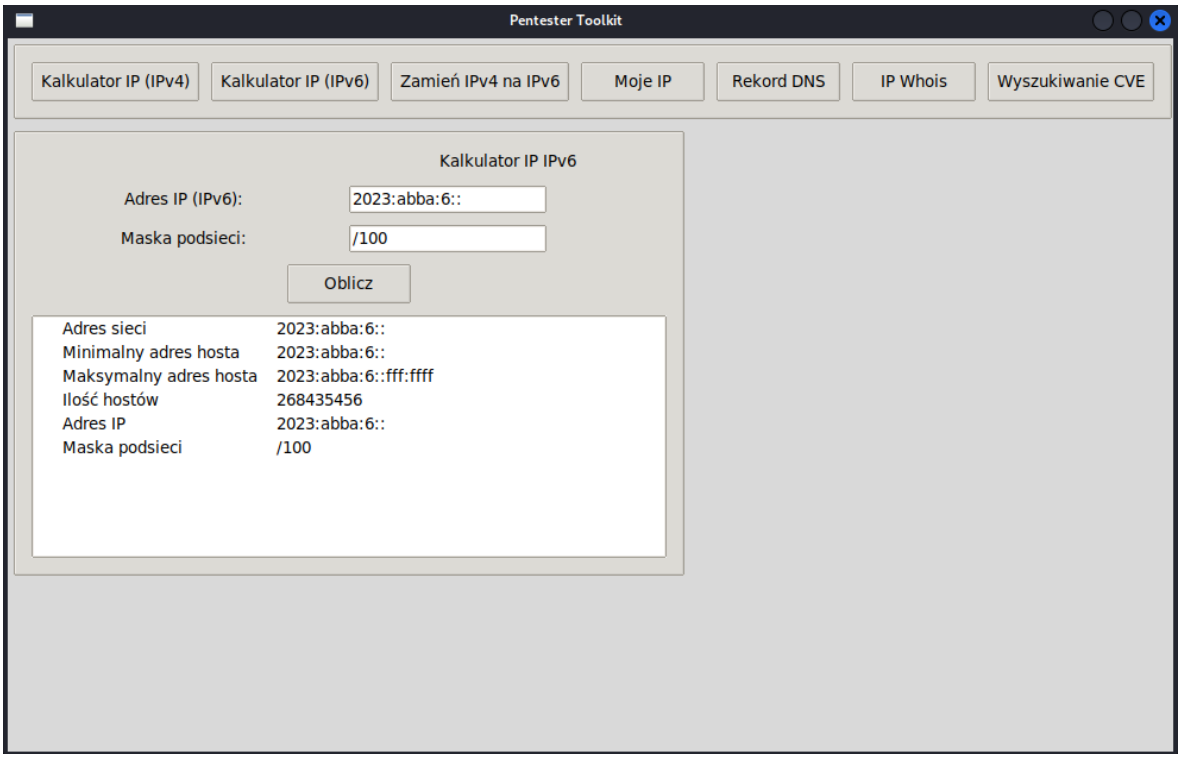

Fig 12. Penetration tester toolkit – printscreen no 4

The entire program was written using Python, using several popular libraries, where *Tkinter* provides the graphical interface. The program also uses various free APIs to obtain and verify various data, such as queries to the CVE vulnerability database. The features it offers include an IPv4 and IPv6 calculator, which allows to calculate data such as the number of addresses in a subnet, network address, and so on. It can also search for DNS records, IP address, or mnemonic data, and vulnerabilities in the database using CVE identifiers. It can

also display a public IP address along with related data. Due to the program structure, it can be easily expanded with additional features.

In summary, many of these features can be found on various websites and as add-ons to programs. They have been brought together here in this program to make their usage faster, more efficient, and easier.

Results

The tools created for penetration testing are often unique because they are designed for a specific purpose. This is not the case for highly complex solutions developed over years by groups of individuals or organizations, such as the *Metasploit Framework* or *Nessus*. These have a wide range of applications, but have been developed with considerable effort and over a relatively long period of time. Less complex programs, such as us scripts, focus on faster execution and do not require knowledge of the tool due to their clear and intuitive graphical interface. We selected very popular and frequently used applications for these tools so that they can be useful in a wide variety of penetration tests.

The script for host scan and brute force password attacks uses *nmap* to search for hosts and scan their ports. It is one of the best tools for such tasks. For brute force password attack, we use the capabilities of tools such as *Hydra*, *Medusa*, and *Ncrack*, which are also very good and efficient solutions. In addition, we created a graphical interface that simplifies their use. This combination makes it an efficient, effective, and easy-to-use solution that can be used successfully by many people, including those unfamiliar with the syntax required to use these tools.

The custom packet generation tool is a very fast tool without redundant options that often complicate first-time users. Several packet creation tools exist, such as *hping*, *Nemesis*, and *Colasoft Packet Builder*. The first two are command-line-based, while the third has a graphical interface that is extensive but therefore more challenging to use initially. Many people do not need the more advanced features they offer. Our program allows quick and transparent packet creation and sending. Of course, if you need to generate and send a very detailed packet, more complex solutions are a better solution. However, our goal was to create a tool that is fast, easy to use, and has a graphical interface. Furthermore, its modular design allows for future expansion at a very low cost.

The third solution we created is a suite of tools that are useful for a wide variety of tasks, including penetration testing. The capabilities of this program are nothing extraordinary, but they are often found separately. They can be found on various websites and in many programs as add-ons. Using them often involves using a browser and unknown servers. In this solution, we used proven servers and implemented proven and well-known Python libraries. The tools in this program will make testing less time-consuming. These include an IP address calculator for versions 4 and 6, checking a public IP address, resolving DNS records, checking an IP address or domain in the Whois database, and searching the CVE vulnerability database. All of these capabilities are offered by many different applications, but here they are combined in a single tool with a clear and easy-to-use graphical interface. Furthermore, this solution allows for an easy expansion of the suite with new capabilities in the future.

Conclusions

All three tools we created were primarily intended to streamline and automate the penetration testing tasks they addressed. By creating a user-friendly graphical interface for each, we aimed to make their use as easy and quick as possible. This graphical environment allows for use without knowledge of command syntax or the command line. These tools can be a good choice for beginners looking to develop in the field of security, specifically penetration testing. The clear, simple, and straightforward interface makes entry easy for potential users and eliminates barriers to starting. The entire tool was created in the widely popular Python language, which facilitates the program's continued improvement and development.

In summary, these tools can be useful during various phases of penetration testing. For example, the first program can be used successfully during discovery, the second during penetration testing, and the third during planning. The demand for tools, programs, and scripts is very high. Many problems can be solved more easily, faster, and more efficiently. They can help solve many penetration testing issues more effectively through automation or more effective algorithms. The tools we have created represent only a small part of the challenges facing the ever-evolving field of penetration testing. Every solution ever written represents only a small part of

what is needed to conduct penetration testing. The appropriate use of tailored tools ensures effective and accurate penetration testing.

However, because of their modular design, the prepared tools can be easily expanded to support additional protocols in the future. It is worth to mention that this work constitutes only the first step in becoming familiar with writing such programs that help to understand how and what is working in the penetration tests environment and network.

Acknowledgment

This paper was supported from the funds of the Ministry of Science and Higher Education, Poland for the year 2025 under Grant 0313/SBAD/1315.

References

- The DMitry official website, https://github.com/jaygreig86/dmitry, last access: Dec 16, 2024.
- Travis, *The Harvester: A Tool For Gathering Email Accounts User Names And Hostnames/Subdomains*, https://www.systranbox.com/how-to-use-theharvester-in-kali-linux/, 2022, last access: Apr 17, 2025.
- The Hashcat official website, https://hashcat.net/wiki/, last access: Apr 12, 2025.
- Ed Moyle, *How to use the Hydra password-cracking tool*, https://www.techtarget.com/searchsecurity/tutorial/How-to-use-the-Hydra-password-cracking-tool, 2022, last access: Apr 6, 2025.
- The John the Ripper official website, https://www.openwall.com/john/doc/, last access: Apr 6, 2025.
- The Kali official website, https://www.kali.org/, last access: Apr 14, 2025.
- The Legion official website, https://github.com/GoVanguard/legion, last access: Apr 12, 2025.
- The Maltego official website, https://docs.maltego.com/support/home, last access: Apr 6, 2025.
- The Medusa official website, http://foofus.net/goons/jmk/medusa/medusa.html, last access: Apr 12, 2025.
- The Mimikatz official website, https://github.com/gentilkiwi/mimikatz/wiki, last access: Apr 6, 2025.
- The Ncrack official website, https://nmap.org/ncrack/man.html, last access: Apr 12, 2025.
- Official nmap website, https://nmap.org/man/pl/, last access: Apr 6, 2025.
- The Ophcrack official website, https://sourceforge.net/p/ophcrack/wiki/Frequently\%20Asked\%20Questions/, last access: Apr 7, 2025.
- The OSRFramework official website, https://github.com/i3visio/osrframework, last access: Apr 12, 2025.
- The Raspberry Pi official website with software to download, https://www.raspberrypi.com/software/, last access: Apr 14, 2025.
- The Recon-ng official website, https://github.com/lanmaster53/recon-ng, last access: Apr 16, 2025.
- J.M. Porup, *What is Shodan? The search engine for everything on the internet*, https://www.csoonline.com/article/3276660/what-is-shodan-the-search-engine-for-everything-on-the-internet.html, 2022, last-access: Apr 10, 2025.
- The Spiderfoot official website, https://github.com/smicallef/spiderfoot, last access: Apr 7, 2025.
- The Wireshark official website, https://www.wireshark.org/docs/wsug_html_chunked/, last access: Apr 6, 2025.