

# Narzędzia do testów cyberbezpieczeństwa w ŚRODOWISKU KALI LINUX

PROJEKT GRUPOWY 2022/2023

Kamil Czepiel  
Jakub Grzybowski  
Jakub Pluta  
Arkadiusz Sałata

Politechnika Gdańska, Wydział ETI, KSSR



## SPIS TREŚCI

1.	WPROWADZENIE .....	5
1.1.	ABSTRAKT .....	5
1.2.	O KALI LINUX.....	5
1.3.	WYKORZYSTANE NARZĘDZIA.....	7
2.	SŁOWNIK TERMINÓW .....	8
3.	NARZĘDZIA DO TESTÓW BEZPIECZEŃSTWA APLIKACJI INTERNETOWYCH .....	9
3.1.	Burp Suite .....	9
	Protokół HTTP.....	9
	Uruchomienie .....	9
	Jak zacząć korzystać?.....	11
	Target .....	13
	Proxy.....	14
	Intercept .....	14
	HTTP history .....	15
	Websockets history .....	16
	Options .....	16
	Intruder.....	18
	Positions .....	19
	Payloads.....	20
	Resource Pool.....	20
	Options .....	21
	Repeater .....	21
	Decoder .....	22
	Comparer.....	22
	Logger .....	24
	Extender .....	24
	Przykład: Szukanie podatności SQL Injection .....	24
4.	NARZĘDZIA DO ŁAMANIA I TESTOWANIA HASEŁ.....	28
4.1.	John The Ripper .....	28
	Najważniejsze komendy .....	28
	Czym tak naprawdę jest hash? .....	28
	Dictionary attack (atak słownikowy) .....	29
	Brute-force attack (atak siłowy) .....	29
	Uruchomienie .....	29

Atakowanie haseł formatu MD5 – dictionary attack .....	30
Atakowanie haseł formatu MD5 – brute-force .....	34
Maskowanie .....	34
Atakowanie haseł z plików jpg, pdf – dictionary attack .....	35
Atakowanie plików zip – dictionary attack.....	36
<b>5. NARZĘDZIA DO ZABEZPIECZANIA POŁĄCZEŃ SIECIOWYCH.....</b>	<b>37</b>
<b>5.1. Nmap .....</b>	<b>37</b>
Skanowanie sieci na bazie protokołu TCP .....	37
Skan TCP .....	37
Skan UDP .....	38
Skan SYN .....	38
Skan ACK.....	38
Skan FIN.....	38
Inne typy skanowania sieci.....	38
Komendy do skanowania sieci .....	39
Skanowanie sieci po „cichu” .....	39
Przykłady skanowania z innymi przełącznikami .....	40
Wykrywanie systemu operacyjnego hosta oraz wersji usług.....	41
Test skanowania agresywnego.....	41
<b>6. NARZĘDZIA DO PRZEPROWADZANIA AUDYTÓW BEZPIECZEŃSTWA.....</b>	<b>43</b>
<b>6.1. Lynis .....</b>	<b>43</b>
Środowisko pracy z oprogramowaniem.....	43
Instalacja.....	44
Obszary działania narzędzia oraz pierwsze uruchomienie .....	45
Efekt pracy – raport ogólny .....	45
Szczegółowy wynik działania programu .....	46
Rozruch.....	47
Uprawnienia do plików.....	48
Informacje o sieci .....	48
Dodatkowe tryby i testy .....	48
Korzystanie z narzędzia Lynis bez instalacji.....	51
Podsumowanie .....	52

# 1. WPROWADZENIE

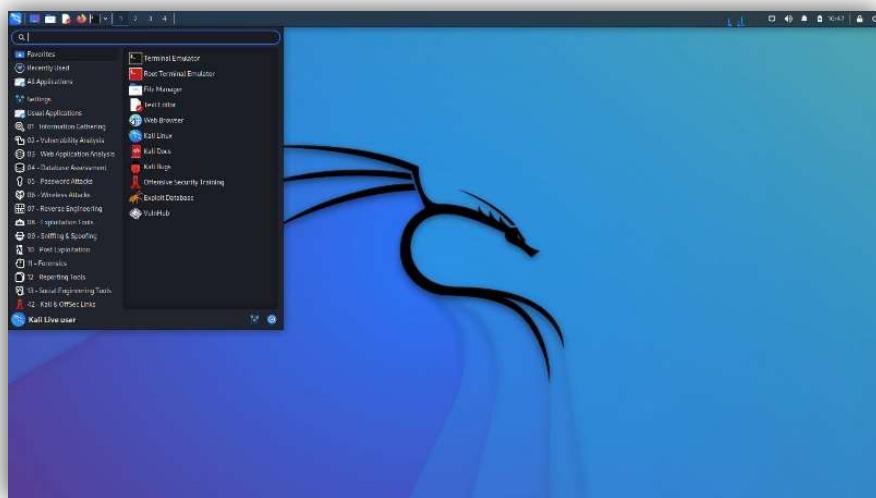
## 1.1. ABSTRAKT

Treścią niniejszego dokumentu jest prezentacja narzędzi do testowania cyberbezpieczeństwa w środowisku Kali Linux w ramach Projektu Grupowego na trzecim roku studiów inżynierskich na Politechnice Gdańskiej na kierunkach Informatyka oraz Telekomunikacja prowadzonych na Wydziale Elektroniki Telekomunikacji i Informatyki. Autorzy dokumentu dokonali szeregu eksperymentów oraz testów owych narzędzi, których funkcjonalność oraz przypadki użycia są opisane w kolejnych rozdziałach. Dokument, zwany dalej manuałem, dokumentacją lub poradnikiem jest stworzony w celach edukacyjnych, kierowany jest do osób, które interesują się cyberbezpieczeństwem i stawiają swoje pierwsze kroki tej dziedzinie. Podawana wiedza ma charakter czysto praktyczny, jednakże do każdego użytego narzędzia zamieszczona jest część teoretyczna, która umożliwia lepsze zrozumienie tematyki oraz ułatwia korzystanie z tychże narzędzi.

Każde test został przeprowadzany w sposób niezagraczący innym użytkownikom sieci oraz w trosce o bezpieczeństwo każdego z nich. Pokazywane przykłady są w pełni zasymulowane na wirtualnych środowiskach w obrębie własnych maszyn komputerowych oraz sieciowych.

## 1.2. O KALI LINUX

Kali Linux to dystrybucja systemu operacyjnego oparta na **jądrze Linux** bazująca na dystrybucji **Debian**. Pierwsze wydanie Kali miało miejsce w 2013 roku. To wydanie od początku było przeznaczone do użytku pod kątem badania **cyberbezpieczeństwa**, wykonywania testów penetracyjnych bądź zabezpieczeń sieciowych. Kali zawiera wiele pre-instalowanych **narzędzi** o otwartym kodzie (ang. **open-source**) oraz pakietów, które spełniają te zadania. Użytkownik na starcie ma możliwość wyboru jak bardzo jego system ma być „opakowany” w owe aplikacje: istnieje wersja *Standard* która zawiera jedynie podstawowe programy oraz wersja *Everything*, które – jak zapewnia producent – zawiera „wszystkie możliwe narzędzia”.

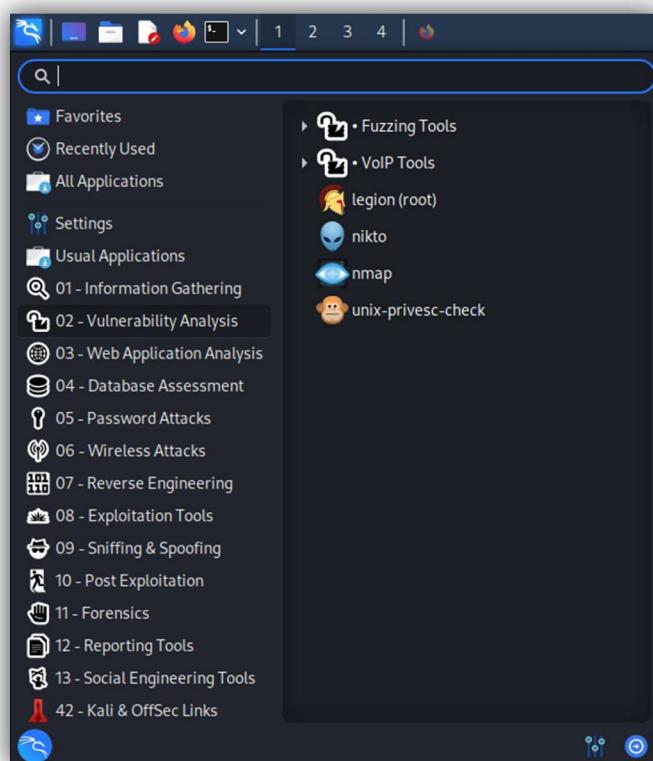


Ryc. 1 – pulpit w Kali Linux

Kali jest również dostępny w wersji *Live*, do której działania nie potrzeba instalacji (system jest wgrywany bezpośrednio z nośnika zewnętrznego typu pendrive USB lub płyta CD/DVD). Producent oferuje pobranie już „zbudowanego” systemu w celu utworzenia maszyny wirtualnej na różnych platformach do wirtualizacji, w tym *VmWare* czy *VirtualBox*. Do tego wszystkiego dochodzą wydania mobilne, chmurowe oraz kontenerowe. **Różnorodność** typów Kali Linux jest **ogromna**, stąd ułatwia to pracę w różnych warunkach, na różnym sprzęcie i przy różnych wymaganiach.

Aktualnie Kali zawiera ponad **600 aplikacji**, które wspierają testy penetracyjne. Szczegółową ich listę można znaleźć w Kali Tools. Mnogość funkcji wymaga wnikliwego zapoznania z systemem i wielogodzinnej praktyki, która pozwoli w pełni wykorzystać cały jego potencjał. Co ważne, system jest całkowicie bezpłatny. W dodatku ma przyjazny dla użytkownika interfejs, a jego instalacja jest banalnie prosta. Ze względu na jego globalne wykorzystanie, wsparcie oprogramowania jest dostępne w wielu językach. A najlepsze jest to, że Kali można także dostosowywać do indywidualnych upodobań. Linux dopasowuje się do oczekiwanych współczesnych użytkowników. Nie można jednak oczekwać, że losowo dodawane repozytoria i pakiety, które są poza standardowymi źródłami oprogramowania, będą działać prawidłowo. Na pewno każda zmiana wymaga nieco kombinacji i wysiłku od strony zainteresowanego wprowadzeniem własnej zmiany. Jednak mimo tego, użytkownicy niezwykle cenią sobie pewną dозę elastyczności, na którą mogą sobie pozwolić w ramach użytkowania tego systemu.

Warto wspomnieć, że Kali jest dystrybucją, która działa według idei „**ofensywnego bezpieczeństwa**” (**Offensive Security**) i to czyni ją jednym z ważniejszych systemów Linuksa. Zabezpieczanie oprogramowania w tym ujęciu polega na dokonywaniu systematycznych prób obejścia zabezpieczeń i dostania się do środka. A to z kolei pozwala na wzmocnienie słabych punktów i usunięcie luk, które mogłyby uszkodzić zewnętrzny atak hakerski.



Ryc. 2 – zakładki z narzędziami do testów cyberbezpieczeństwa

### 1.3. WYKORZYSTANE NARZĘDZIA

Poniżej znajduje się lista oprogramowania, które zostało użyte do sporządzenia tej dokumentacji w kolejnych rozdziałach:

**BeEF** (The Browser Exploitation Framework) – pakiet do testowania zabezpieczeń w obrębie przeglądarki, głównie jej skryptów; badanie podatności na ataki typu **XSS** (Cross-Site Scripting) oraz **CSRF** (Cross-Site Request Forgery);

**Burp suite** – narzędzie do przechwytywania żądań **HTTP** oraz ich modyfikacji w trakcie przesyłania do serwera;

**Hydra** – narzędzie do testowania siły haseł oraz ich łamania atakami **brute-force**;

**John the Ripper** – narzędzie do testowania haseł w formie **hashów** (MD4, MD5, Kerberos ASF oraz inne) za pomocą **brute-force** oraz **metodami słownikowymi**;

**Lynis** – narzędzie konsolowe do wykrywania luk w zabezpieczeniach serwerów oraz monitorowania ich aktualnego stanu zabezpieczeń. Narzędzie to można również wykorzystać do audytowania dowolnej maszyny, którą ten program wspiera.

**Nmap** – program do analizowania sieci poprzez protokoły TCP oraz inne; sprawdzanie otwartych portów, połączonych hostów, komunikacji z nimi. Program zawiera wiele wbudowanych skryptów, dzięki którym można uzyskać wiele informacji o podanej sieci;

**Nikto** – pakiet do skanowania sieci w celu odkrycia podatności serwerów i stron internetowych. Pozwala wykryć przestarzałe oprogramowanie czy szkodliwe pliki.

## 2. SŁOWNIK TERMINÓW

Korzystanie z oprogramowania do badania cyberbezpieczeństwa wiąże się z koniecznością szerszego zapoznania się z techniczną nomenklaturą obowiązującą w tej dziedzinie. Toteż poniżej podajemy większość z istotnych terminów i haseł, które przydadzą się przy okazji obcowania z tym dokumentem.

### A

**aplikacja** – program (biznesowy / dla zwykłych użytkowników; tekstowy / graficzny), który ma określone zadania w systemie komputerowym.

**atak (hakerski)** – działanie, które ma na celu włamanie do systemów komputerowych poprzez wykorzystanie luk w zabezpieczeniach tychże systemów.

### B

**back-end** – część logiczna aplikacji; zestaw funkcji, usług i modułów, które stanowią logikę biznesową aplikacji.

**burp** – proces restartu sprzętu sieciowego, jego usług lub inne przerwania w sieci komputerowej (np. utrata pakietów z danymi)

### C

**ciasteczka (cookies)** – dane niewielkich rozmiarów zwracane przez serwer i przechowywane w przeglądarce, wysyłane z każdym żądaniem.

### D

**domena** – unikalny ciąg znaków, który jest tłumaczony na adres ip przez serwery DNS.

### E

**encja** – obiekt świata rzeczywistego przedstawiony jako zbiór atrybutów (cech), który go identyfikują.

### F

**front-end** – część wizualna aplikacji; to jak użytkownik ją widzi poprzez graficzny interfejs; zestaw funkcji i modułów odpowiadający za prezentację aplikacji.

**fuzzing** – sposób testowania oprogramowania poprzez dostarczanie nieprawidłowego wejścia i sprawdzanie jak program na nie reaguje.

### L

**localhost** – standardowa nazwa hosta, która jest tłumaczona na adres lokalnej maszyny. Innym słowy adres wskazujący na aktualną maszynę.

### P

**proxy** – serwer pośredniczący w komunikacji pomiędzy klientem, a serwerem.

### 3. NARZĘDZIA DO TESTÓW BEZPIECZEŃSTWA APLIKACJI INTERNETOWYCH

#### 3.1. Burp Suite

Burp Suite to narzędzie z graficznym interfejsem użytkownika wykorzystywane w dziedzinie bezpieczeństwa aplikacji internetowych. Występuje w trzech wersjach:

- Community Edition
- Professional Edition
  - Pozwala zapisać stan pracy
- Enterprise Edition

#### Protokół HTTP

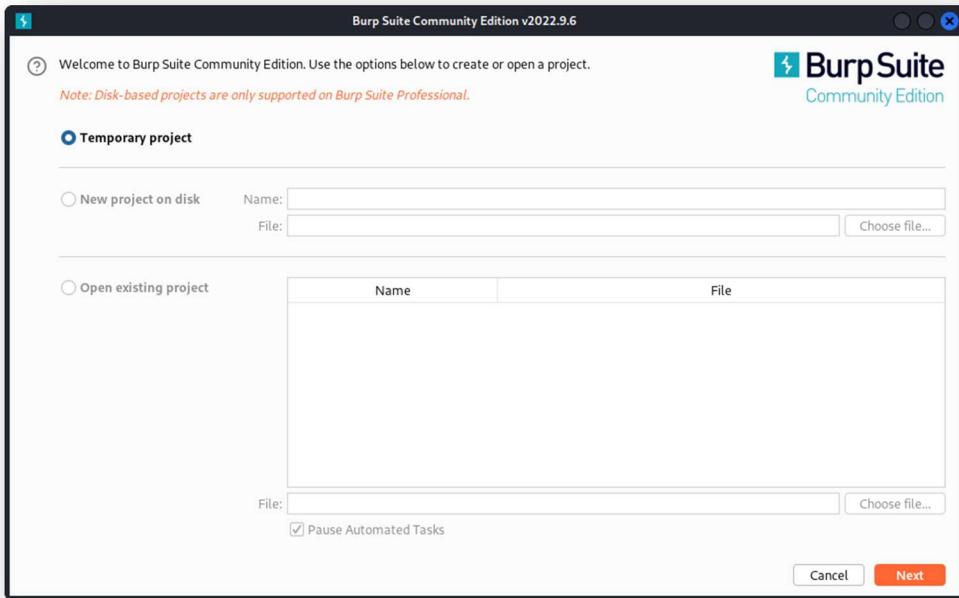
HTTP (ang. Hypertext Transfer Protocol) to protokół określający reguły przesyłania zasobów i zasady komunikacji na drodze klient - serwer. Protokół HTTP definiuje znormalizowany sposób w jakim informacje są udostępniane, przetwarzane i odczytywane przez serwer oraz jak wygląda odpowiedź na żądania. Żądanie http składa się z nagłówków oraz ciała (niewymagane). Nagłówki zawierają wiele istotnych informacji, które serwer wykorzystuje do interpretacji danych.

Żądania http mogą mieć różnych cel od odczytu danych, ich modyfikację, wstawienie nowych do usunięcia. O tym jakie zadanie pełni żądanie mówi nam metoda. Metoda http jest wiele, ale do najpopularniejszych które warto poznać i być świadomym należą:

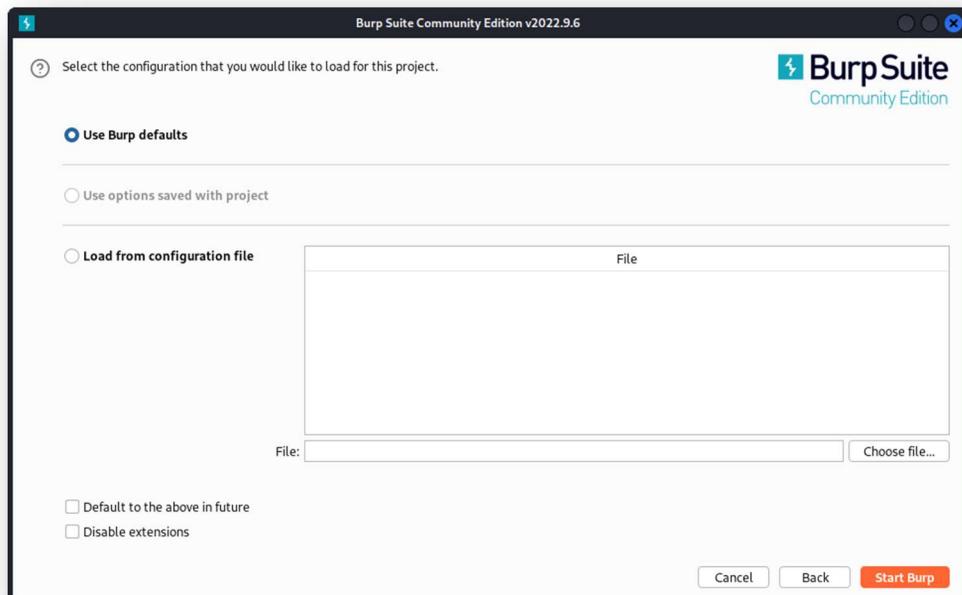
- GET – pobranie zasobu wskazanego przez URI, może mieć postać warunkową jeśli w nagłówku występują pola warunkowe takie jak "If-Modified-Since"
- HEAD – pobiera informacje o zasobie, stosowane do sprawdzania dostępności zasobu
- PUT – przyjęcie danych przesyłanych od klienta do serwera, najczęściej aby zaktualizować wartość **encji**
- POST – przyjęcie danych przesyłanych od klienta do serwera (np. wysyłanie zawartości formularzy)
- DELETE – żądanie usunięcia zasobu, włączone dla uprawnionych użytkowników

#### Uruchomienie

Po uruchomieniu programu, pierwszym widokiem jaki zostaje nam udostępniony jest widok wyboru projektu. Wersja community ogranicza nasz wybór do projektów tymczasowych, gdzie po zamknięciu programu nasze prace zostaną utracone. Dla naszych zastosowań jak najbardziej to wystarczy, jednak jako profesjonalista będziemy chcieli wykupić licencję.



Następnie mamy możliwość użycia domyślnej konfiguracji lub załadowania jej z pliku. Możemy chcieć skorzystać z tej opcji, gdy pracujemy dla różnych klientów i każdy z nich wymaga innej konfiguracji.



Po wybraniu konfiguracji naszym oczom ukazuje się panel główny programu.

Pasek zakładek jest czymś na co powinniśmy zwrócić szczególną uwagę. Każda zakładka odpowiada to innemu narzędziu. Dostępne zakładki to:

- Dashboard
- Target

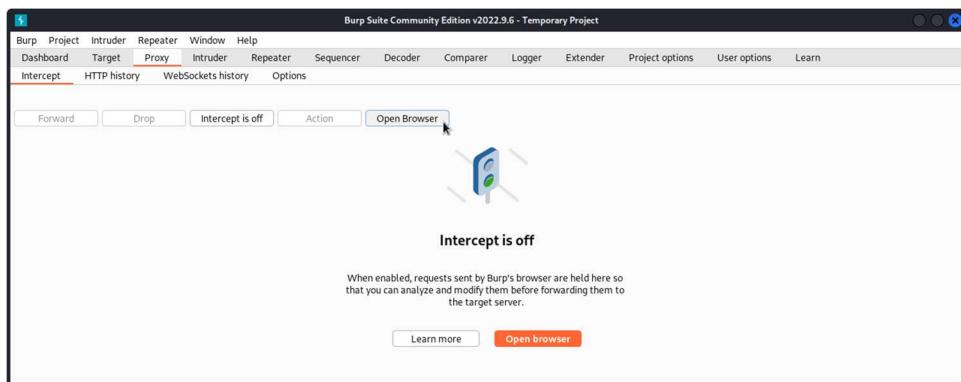
- Proxy
- Intruder
- Repeater
- Sequencer
- Decoder
- Comparer
- Logger
- Extender
- Project options
- User options
- Learn

Każdej z nich przyjrzymy się dokładnie w dalszej części rozdziału.

Jak zacząć korzystać?

Burp Suite powinien pośredniczyć w komunikacji pomiędzy aplikacją **frontendową** działającą w przeglądarce, a aplikacją **backendową** działającą na serwerze. Żeby przekierować ruch z przeglądarki do Burp'a zamiast od razu do docelowego serwera możemy skorzystać z rozszerzeń do przeglądarki np. **FoxyProxy** lub korzystając z już skonfigurowanej przeglądarki udostępnionej przez Burp'a, którą możemy uruchomić z zakładki

### Proxy > Intercept



Dzięki temu wszystkie żądania jakie wykona przeglądarka zostaną wylistowane w zakładce:

### Proxy > HTTP history

Po uruchomieniu przeglądarki udało się na naszą demonstracyjną stronę bWapp (hostowaną na lokalnym serwerze), gdzie przeszedłem proces logowania. Będąc ciekawym co nt. cyberbezpieczeństwa ma do powiedzenia Wikipedia, wyszukałem tam wymienioną frazę. Teraz mam dostęp do wszystkich żądań.

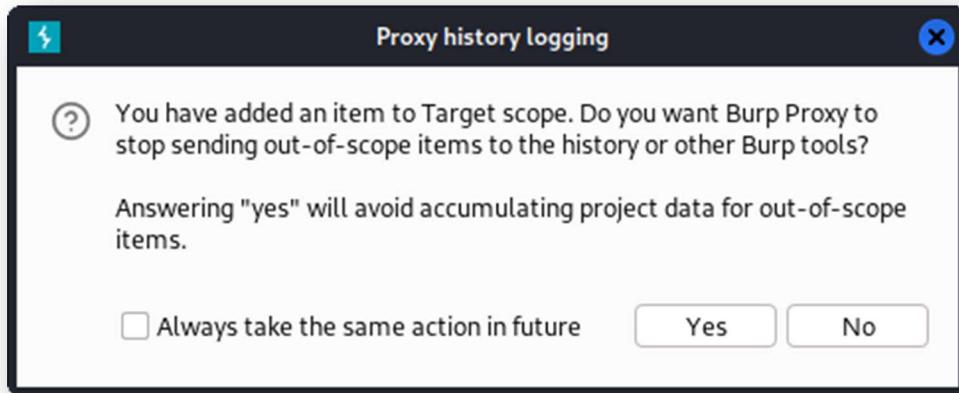
The screenshot shows the Burp Suite interface with the 'Proxy' tab selected. The 'HTTP history' section displays a list of requests. One specific request, 'http://localhost/login.php', is highlighted with an orange background. The 'Request' and 'Response' panes below show the details of this selected request.

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment
188	https://pl.wikipedia.org	GET	/api/rest_v1/page/summary/Wikipedia		✓	200	4648	JSON			
189	https://pl.wikipedia.org	GET	/w/load.php?lang=pl&modules=query... /w/load.php?lang=pl&modules=ext.uls...		✓	200	25292	script	php		
190	https://pl.wikipedia.org	GET	/w/load.php?lang=pl&modules=ext.uls...		✓	200	148005	script	php		
191	http://localhost	GET	/login.php		✓	200	4363	HTML	php	bWAPP - Login	
192	http://localhost	POST	/login.php		✓	302	502	HTML	php		
193	http://localhost	GET	/portal.php		✓	200	23702	HTML	php	bWAPP - Portal	
194	https://passwordsleakcheck.p... a.com	POST	/V1/leaks/lookupSingle		✓	400	639	script	php		
195	https://pl.wikipedia.org	GET	/w/load.php?lang=pl&modules=ext.uls...		✓	200	38515	script	php		
196	https://pl.wikipedia.org	GET	/w/extensions/UniversalLanguageSelect...		✓	200	3326	XML	svg		
197	https://pl.wikipedia.org	GET	/w/extensions/UniversalLanguageSelect...		✓	200	2069	XML	svg		
198	https://pl.wikipedia.org	GET	/w/api.php?action=cirus-config-dump...		✓	200	1365	JSON	php		
199	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2194	JSON	php		
200	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2177	JSON	php		
201	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2177	JSON	php		
202	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2179	JSON	php		
203	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2179	JSON	php		
204	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2190	JSON	php		
205	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	2263	JSON	php		
206	https://pl.wikipedia.org	GET	/w/api.php?action=opensearch&format...		✓	200	1608	JSON	php		

Nie zawsze jednak będziemy chcieli przechwytywać żądania z każdej strony, z jakiej korzystamy. W celu zawężenia listy żądań skorzystamy z zakładki **Target**. Spośród wszystkich **domen** do których zgłaszały się po zasoby wybieramy tą która nas interesuje, w tym przypadku będzie to **localhost**, na którym serwowana jest nasza aplikacja.

The screenshot shows the Burp Suite interface with the 'Scope' tab selected. In the left sidebar, under the 'Site map' section, there is a tree view of domains and hosts. An item 'http://localhost/' has a context menu open above it. The menu includes options like 'Add to scope', 'Scan', 'Engagement tools [Pro version only]', 'Compare site maps', 'Expand branch', 'Expand requested items', 'Collapse branch', 'Delete host', 'Copy URLs in this host', 'Copy links in this host', 'Save selected items', 'Show new site map window', and 'Site map documentation'. The 'Add to scope' option is highlighted with a mouse cursor.

Burp zapyta nas, czy pomijać żądania, które nie należą do zakresu (**scope**). Jeśli test bezpieczeństwa chcemy przeprowadzić na konkretnej stronie jak najbardziej chcemy kliknąć **Yes**.



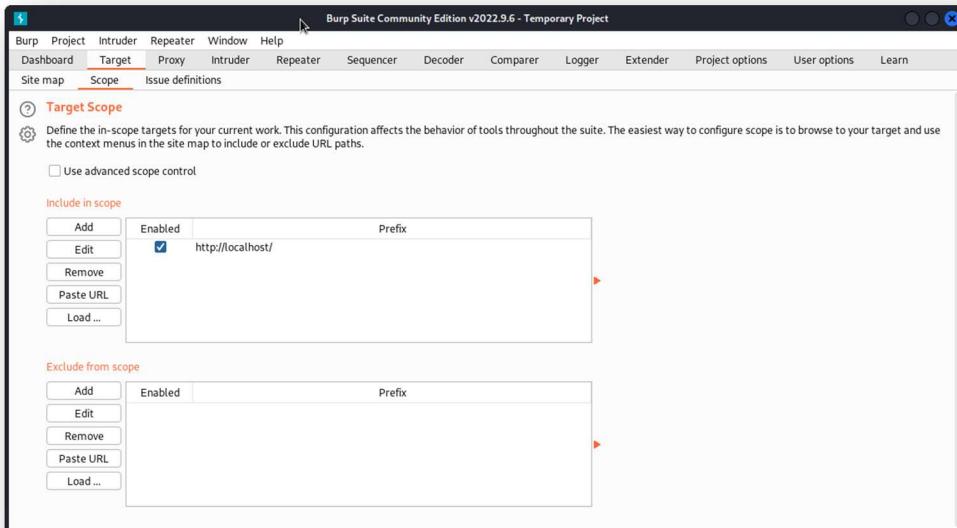
Teraz jesteśmy gotowi do szukania podatności.

## Target

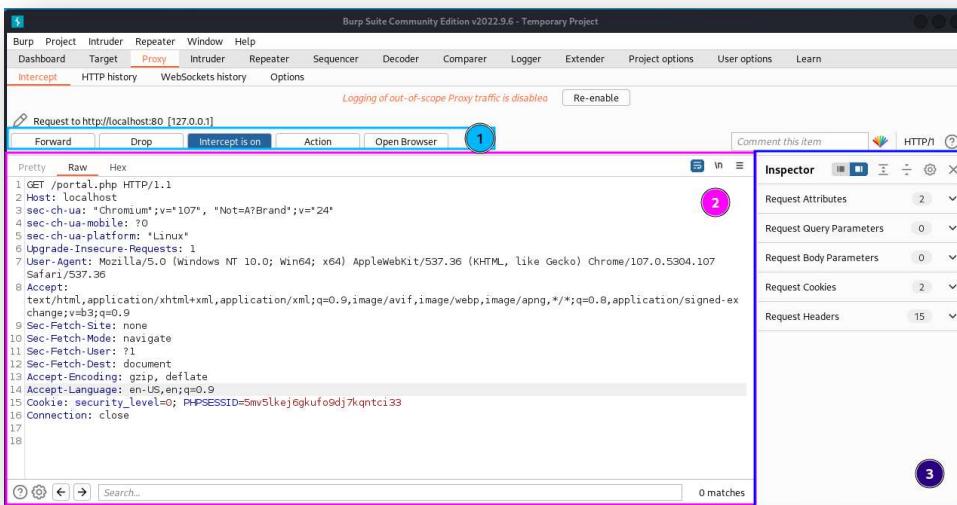
Narzędzie zawierające między innymi mapę strony (1) (ang. **site map**), czyli strukturę strony jaką udało się ustalić burpowi na podstawie żądań jakie przez niego przeszły. Każdy węzeł strony (może to być obraz, styl **CSS** lub plik **PHP**) zawiera listę żądań (2) na podstawie których został ustalony. Każde żądanie i odpowiedź możemy podejrzeć (3).

Dodatkowo zakładka pozwala nam zdecydować jakie strony internetowe mają znajdować się w naszym zakresie (ang. **scope**). Możemy to zrobić klikając prawy przycisk myszy (**PPM**) na węzeł, wtedy możemy dodać lub wykluczyć z zakresu lub zrobić to ręcznie w zakładce:

**Target > Scope**



## Proxy Intercept



### Przyciski akcji (1)

- **Forward** – przepuść żądanie dalej, żeby trafiło na serwer.
- **Drop** – porzuć żądanie, żeby nie trafiło na serwer.
- **Intercept is on / Intercept is off** – przełącznik jednocześnie informujący czy przychwytywanie jest włączone.
- **Action** – akcje, które możemy wykonać na żądaniu (podobnie jak PPM na wylistowanym żądaniu w HTTP history).
- **Open Browser** – otwarcie wstępnie skonfigurowanej przeglądarki (**proxy**, **certyfikaty**).

**Żądanie (2)** – podgląd żądania, które zostało wysłane przez przeglądarkę.

**Inspektor (3)** – jeśli interesuje nas konkretna część żądania lub odpowiedzi możemy skorzystać z inspektora, który w przejrzysty sposób nam je wyświetli.

Korzystaliśmy już z tej zakładki, żeby otworzyć wstępnie skonfigurowaną przeglądarkę. Poza tym w zakładce mamy możliwość przychwytywania żądań. Czyli każde żądanie, które znajduje się w naszym zakresie, zostanie tutaj przekierowane i wstrzymane do momentu przepuszczenia żądania dalej (przycisk **Forward**) lub je odrzucić (przycisk **Drop**). Co ważne możemy dowolnie zmodyfikować żądanie według naszych potrzeb np. podmienić **PHPSESSID** (identyfikator sesji, który pozwala serwerowi zidentyfikować, który użytkownik wysłał żądanie).

Z racji, że strony internetowe potrafią wysyłać bardzo dużo żądań, szczególnie podczas ładowania pierwszej strony co wynika z faktu, że pobierane są też style.css, kod **JavaScript** itp. przychwytywanie w większości przypadków będziemy mieć wyłączone lub będziemy mieć skonfigurowane filtry w zakładce **Options**.

## HTTP history

Burp Suite Community Edition v2023.6 - Tempresey Project

Burp History - WebSockets History Options

Request Response Inspector

1 2 3 4

**Żądanie (1)** – podgląd żądania, które zostało wysłane przez przeglądarkę.

**Odpowiedź (2)** – odpowiedź, którą otrzymaliśmy na żądanie z punktu 1.

**Lista żądań (3)** – lista wszystkich żądań przechwyconych przez burp'a.

**Inspektor (4)** – jeśli interesuje nas konkretna część żądania lub odpowiedzi możemy skorzystać z inspektora, który w przejrzysty sposób nam je wyświetli.

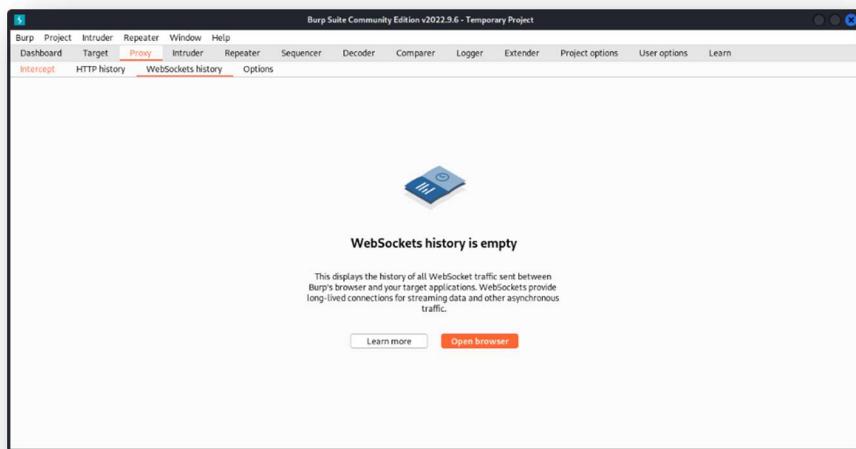
Wylistowane żądania zostały już wysłane, a odpowiedzi otrzymane, więc nie możemy tutaj za dużo zrobić poza przeanalizowaniem żądań. Jeśli jednak któreś przykuło naszą uwagę możemy je wysłać do **repeatera**, który zostanie opisany w osobnym podrozdziale.

Robimy to poprzez:

**Wybranie żądania > PPM > Send to Repeater (Ctrl + R)**

The screenshot shows the Burp Suite interface. A list of requests is visible at the top, with several items highlighted in orange. A context menu is open over one of the requests, specifically the third item in the list. The menu items include: Remove from scope, Scan, Send to Intruder (Ctrl+I), Send to Repeater (Ctrl+R), Send to Sequencer, Send to Comparer (request), Send to Comparer (response), Show response in browser, Request in browser, Engagement tools [Pro version only], and Show new history window. The 'Send to Repeater' option is highlighted with a cursor.

## Websockets history



Podobnie jak historia HTTP, służy do podglądu asynchronicznej komunikacji poprzez gniazda internetowe czyli **WebSockets**.

## Options

Konfiguracja dotycząca zakładki **Proxy**. Dostępne opcje:

**Proxy listeners** – umożliwia skonfigurowanie **nastuchiwacza**. Nastuchiwacze służą do nasłuchiwanego żądań przychodzących z przeglądarki. Jeśli korzystamy z burpowej przeglądarki nie musimy tutaj nic robić, jeśli natomiast korzystamy np. z Firefox'a powinniśmy w nim skonfigurować proxy, aby przekierowywało ruch na dodany przez nas listener.



**Intercept Client Requests** – pozwala ustawić filtry przechwytywania żądań jeśli opcja **Intercept > Intercept is on** jest włączona.

The screenshot shows the 'Intercept Client Requests' configuration page. It includes a table for defining rules based on file extension, request parameters, HTTP method, and URL. There are also checkboxes for automatically fixing new lines and updating Content-Length headers.

Add	Enabled	Operator	Match type	Relationship	Condition
<input type="button" value="Edit"/>	<input checked="" type="checkbox"/>		File extension	Does not match	(^gif\$ jpg\$ png\$ css\$ js\$ ico\$ ^...
<input type="button" value="Remove"/>		Or	Request	Contains parameters	
<input type="button" value="Up"/>		Or	HTTP method	Does not match	(get post)
<input type="button" value="Down"/>		And	URL	Is in target scope	

Automatically fix missing or superfluous new lines at end of request  
 Automatically update Content-Length header when the request is edited

**Intercept Server Responses** – pozwala ustawić filtry przechwytywania odpowiedzi jeśli opcja **Intercept > Intercept is on** jest włączona.

The screenshot shows the 'Intercept Server Responses' configuration page. It includes a table for defining rules based on content type header, request status code, and URL. There is a checkbox for automatically updating Content-Length headers.

Add	Enabled	Operator	Match type	Relationship	Condition
<input type="button" value="Edit"/>	<input checked="" type="checkbox"/>		Content type header	Matches	text
<input type="button" value="Remove"/>		Or	Request	Was modified	
<input type="button" value="Up"/>		Or	Request	Was intercepted	
<input type="button" value="Down"/>		And	Status code	Does not match	^304\$
		And	URL	Is in target scope	

Automatically update Content-Length header when the response is edited

**Intercept WebSockets Messages** – pozwala wybrać czy chcemy przychwytywać komunikację klient-serwer, serwer-klient czy obie.

The screenshot shows the 'Intercept WebSockets Messages' configuration page. It includes checkboxes for intercepting client-to-server and server-to-client messages.

Intercept client-to-server messages  
 Intercept server-to-client messages

**Response Modification** – pozwala zmodyfikować odpowiedź zanim trafi z powrotem do przeglądarki.

The screenshot shows the 'Response Modification' configuration page. It lists various options for modifying responses, such as unhide hidden form fields, enable disabled form fields, remove input field length limits, remove JavaScript form validation, remove all JavaScript, remove <object> tags, convert HTTPS links to HTTP, and remove secure flag from cookies.

These settings are used to perform automatic modification of responses.  
 Unhide hidden form fields  
 Prominently highlight unhidden fields  
 Enable disabled form fields  
 Remove input field length limits  
 Remove JavaScript form validation  
 Remove all JavaScript  
 Remove <object> tags  
 Convert HTTPS links to HTTP  
 Remove secure flag from cookies

**Match and replace** – pozwala zdefiniować automatyczną podmianę ciągu znaku na inny zdefiniowany ciąg znaku. Możemy to wykorzystać do ukrywania nagłówków czy ich podmiany.

Add	Enabled	Item	Match	Replace	Type	Comment
<input type="checkbox"/>	<input type="checkbox"/>	Request header	^Referer:\$	Accept-Encoding:\$	Regex	Hide Referer header
<input type="checkbox"/>	<input type="checkbox"/>	Response header	Set-Cookie:\$		Regex	Require non-compressed responses
<input type="checkbox"/>	<input type="checkbox"/>	Request header	+Host: foo.example.org\$	Host: bar.example.org	Regex	Ignore cookies
<input type="checkbox"/>	<input type="checkbox"/>	Request header	Origin: foo.example.org	Origin: foo.example.org	Literal	Rewrite Host header
<input type="checkbox"/>	<input type="checkbox"/>	Response header	^Strict-Transport-Security:		Regex	Add spoofed CORS origin
<input type="checkbox"/>	<input type="checkbox"/>	Response header	X-XSS-Protection: 0		Literal	Remove HSTS headers
<input type="checkbox"/>	<input type="checkbox"/>					Disable browser XSS protection

## TLS Pass Through

Add	Enabled	Host / IP range	Port
<input type="checkbox"/>	<input type="checkbox"/>		

Automatically add entries on client TLS negotiation failure

**Miscellaneous** – szczegółowe ustawienia nie związane z konkretną kategorią.

- Miscellaneous
- These settings control some specific details of Burp Proxy's behavior. You can change the default settings here to deal with particular problems or situations.
- Use HTTP/1.0 in requests to server
- Use HTTP/1.0 in responses to client
- Set response header "Connection: close"
- Set "Connection" header on incoming requests when using HTTP/1
- Strip Proxy-\* headers in incoming requests
- Remove unsupported encodings from Accept-Encoding headers in incoming requests
- Strip Sec-WebSocket-Extensions headers in incoming requests
- Unpack gzip / deflate in requests
- Unpack gzip / deflate in responses
- Disable web interface at http://burpsuite
- Suppress Burp error messages in browser
- Don't send items to Proxy history or live tasks
- Don't send items to Proxy history or live tasks, if out of scope

## Intruder

Intruder to narzędzie, które umożliwia nam przeprowadzenie ataków słownikowych. Dostarczając słownik jesteśmy w stanie sprawdzić odpowiedź dla każdego podanego wejścia. Wersja community wprowadza jednak **throttling**, czyli mimo że nasz sprzęt jest w stanie przetwarzać żądania szybciej, zostanie on specjalnie ograniczony w celu spowolnienia wysłania całego słownika.

Wynikiem operacji będzie lista żądań z testowymi wartościami słownika, którą możemy sortować według tego co nas najbardziej interesuje.

3. Intruder attack of http://localhost - Temporary attack - Not saved to project file

Attack Save Columns								
Results	Positions	Payloads	Resource Pool	Options				
Filter: Showing all items								
Request	Position	Payload	Status	Error	Timeout	Length	Comment	
0			302	<input type="checkbox"/>	<input type="checkbox"/>	502		
1	1	Prefixbee	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		
2	1	Prefixwapp	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		
3	1	Prefixwrap	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		
4	2	Prefixbee	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		
5	2	Prefixwapp	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		
6	2	Prefixwrap	200	<input type="checkbox"/>	<input type="checkbox"/>	4430		

Request	Response
Pretty	Raw Hex
<pre> 1 POST /login.php HTTP/1.1 2 Host: localhost 3 Content-Length: 58 4 Cache-Control: max-age=0 5 sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24" 6 sec-ch-ua-mobile: ?0 7 sec-ch-ua-platform: "Linux" 8 Upgrade-Insecure-Requests: 1 9 Origin: http://localhost 10 Content-Type: application/x-www-form-urlencoded 11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36 12 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed -exchange;v=b3;q=0.9 13 Sec-Fetch-Site: same-origin 14 Sec-Fetch-Mode: navigate 15 Sec-Fetch-User: ?1 16 Sec-Fetch-Dest: document 17 Referer: http://localhost/login.php 18 Accept-Encoding: gzip, deflate 19 Accept-Language: en-US,en;q=0.9 20 Cookie: security_level=0; PHPSESSID=9ng4uhlr3caio76nidk2pgsl7 21 Connection: close 22 23 login=Prefixwapp&amp;password=bug&amp;security_level=0&amp;form=submit </pre>	

## Positions

Burp Suite Community Edition v2022.9.6 - Temporary Project

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy **Intruder** Repeater Sequencer Decoder Comparer Logger Extender Project options User options Learn

7 x 8 x + 1 Positions Payloads Resource Pool Options 2

Choose an attack type 3

Attack type: Sniper

start attack

Payload Positions 4

Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.

Target: http://localhost  Update Host header to match target

1 POST /login.php HTTP/1.1  
2 Host: localhost  
3 Content-Length: 51  
4 Cache-Control: max-age=0  
5 sec-ch-ua: "Chromium";v="107", "Not=A?Brand";v="24"  
6 sec-ch-ua-mobile: ?0  
7 sec-ch-ua-platform: "Linux"  
8 Upgrade-Insecure-Requests: 1  
9 Origin: http://localhost  
10 Content-Type: application/x-www-form-urlencoded  
11 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.5304.107 Safari/537.36  
12 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/\*,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9  
13 Sec-Fetch-Site: same-origin  
14 Sec-Fetch-Mode: navigate  
15 Sec-Fetch-Dest: document  
16 Sec-Fetch-Dest: document  
17 Referer: http://localhost/login.php  
18 Accept-Encoding: gzip, deflate  
19 Accept-Language: en-US,en;q=0.9  
20 Cookie: security\_level=\$0; PHPSESSID=\$9ng4uh1r3caie7enidk2pgs175  
21 Connection: close  
22  
23 login=\$beep&password=\$bug&security\_level=\$0&form=\$submit\$

0 matches Clear

6 payload positions

Length: 924

Pasek zakładek żądań (**1**) – umożliwia łatwe przełączanie się pomiędzy różnymi żądaniami.

Pasek zakładek żądania (**2**) – umożliwia przełączanie się pomiędzy różnymi konfiguracjami.

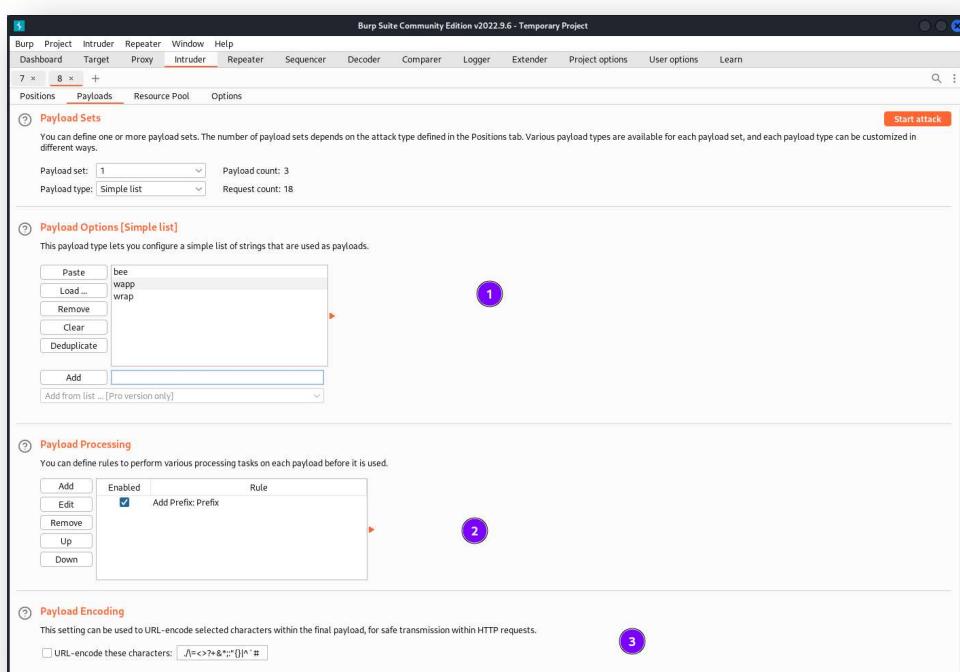
Wybór sposobu w jaki znaczniki będą uzupełniane (**3**) – możliwe do wyboru:

- **Sniper**
- **Battering ram**
- **Pitchfork**
- **Cluster Bomb**

Pozycje znaczników żądania (**4**) – najważniejsza część. Umożliwia ustawienie znaczników, które zostaną zamienione na konkretne wartości podczas **fuzzingu**.

Przyciski dodawania i usuwania słów kluczowych (**5**).

## Payloads



W tej zakładce możemy załadować nasz słownik (**1**), którego wyrazy zostaną wstawione w miejsce znaczników z zakładki **Positions**. Oczywiście im bardziej rozbudowany będzie słownik oraz im bardziej będzie doprecyzowany do danego celu, tym efekt uzyskamy szybciej o ile w ogóle.

Dodatkowo jesteśmy w stanie dodać zasady procesowania każdego słowa (**2**) ze słownika lub zaznaczyć opcję kodowania znaków (**3**).

Opcję kodowania znaków powinniśmy mieć włączoną tylko wtedy, gdy przechwycone żądanie które modyfikujemy będzie zakodowane.

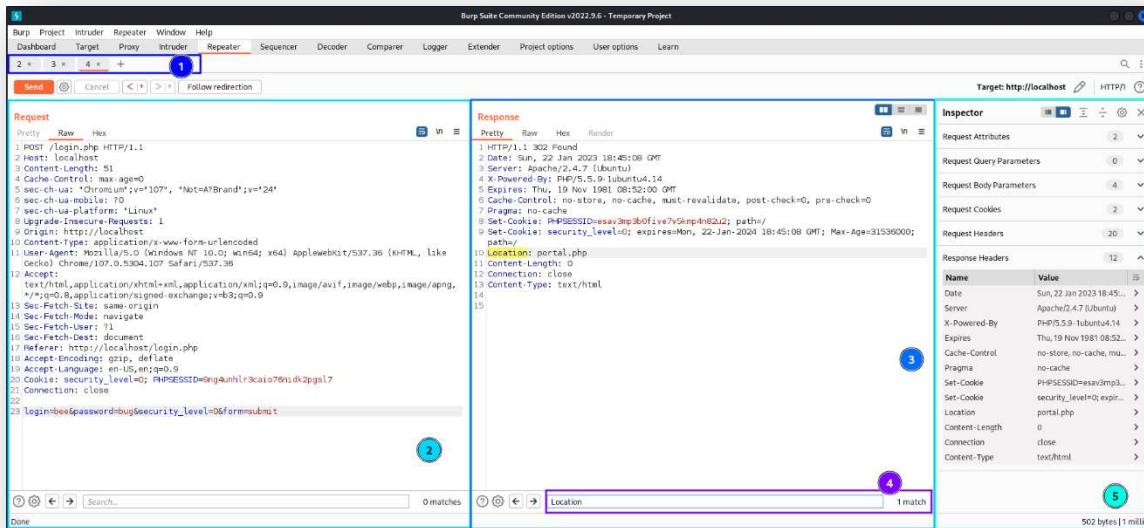
## Resource Pool

Możemy tutaj ograniczyć zasoby, z których będzie korzystać program podczas **fuzzingu**.

## Options

Możemy tutaj skonfigurować jak będzie przebiegać proces. Z najważniejszych opcji możemy skonfigurować liczbę prób w przypadku problemów sieciowych, zdefiniować podświetlanie interesujących słów w odpowiedzi oraz jak rozwiązywać przekierowania.

## Repeater



Pasek zakładek (1) – pozwala łatwo przełączać się pomiędzy żądaniami do powtórzenia

Żądanie (2) – podgląd żądania, które zostało wysłane przez przeglądarkę.

Odpowiedź (3) – odpowiedź, którą otrzymaliśmy na żądanie z punktu 1.

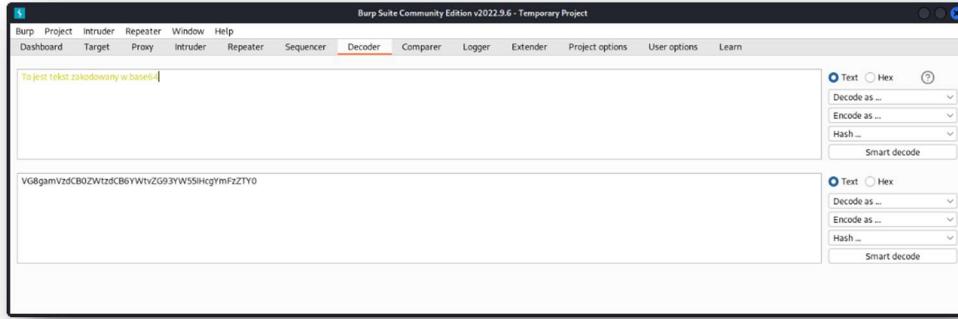
Wyszukiwarka (4) – pozwala wyszukać wpisaną frazę w odpowiedzi.

Inspektor (5) – jeśli interesuje nas konkretna część żądania lub odpowiedzi możemy skorzystać z inspektora, który w przejrzysty sposób nam je wyświetli.

Zakładka pozwala nam eksperymentować z żądaniami, które uznamy za podatne. Otrzymujemy możliwość dowolnej modyfikacji żądania, jego parametrów czy nagłówków i sprawdzenia co na takie żądanie odpowie serwer. Na zrzucie ekranu widzimy żądanie uwierzytelnienia się pod adresem **localhost/login.php** oraz odpowiedź jaką dostaliśmy od serwera. Kod odpowiedzi to **302**, czyli kod z rodziny przekierowań, które powinny zawierać nagłówek **Location**, mówiący przeglądarce gdzie przekierować użytkownika po otrzymaniu odpowiedzi. Burp natomiast pod paskiem zakładek udostępnia nam przycisk **Follow redirection**. Korzystając z wyszukiwarki lub inspektora możemy wywnioskować, że zostaniemy przekierowani na stronę **portal.php**.

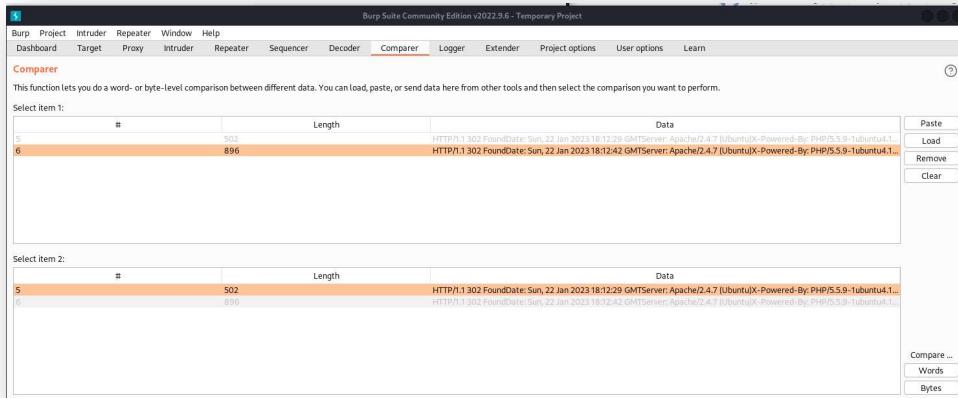
## Decoder

Proste narzędzie do transformacji danych z jednego formatu na drugi. Jest w stanie intelligentnie



rozpoznawać wprowadzone dane. Kodowanie jakie jest obsługiwane:

- Plain
- URL
- HTML
- Base64
- ASCII hex
- Hex
- Octal
- Binary
- Gzip



## Comparer

Proste narzędzie do wizualizacji różnic w danych. Dane możemy wkleić bezpośrednio, załadować z pliku lub przesyłać je z aplikacji np.:

**Proxy > HTTP history > PPM na odpowiedź HTTP > Send to comparer.**

The screenshot shows the Burp Suite interface. At the top, the menu bar includes 'Burp', 'Project', 'Intruder', 'Repeater', 'Window', and 'Help'. Below the menu is a toolbar with 'Dashboard', 'Target', 'Proxy' (which is selected), 'Intruder', 'Repeater', 'Sequencer', 'Decoder', 'Comparer', 'Logger', and 'Extender'. Under 'Proxy', there are sub-options: 'Project options', 'User options', and 'Learn'. The main area is titled 'HTTP history' with tabs for 'Intercept', 'HTTP history' (selected), 'WebSockets history', and 'Options'. A message at the top says 'Logging of out-of-scope Proxy traffic is disabled' with a 'Re-enable' button. A filter bar below says 'Filter: Hiding CSS, image and general binary content'. The main table lists four rows of proxy history:

#	Host	Method	URL	Params	Edited	Status	Length	MIME type
22	http://localhost	POST	/login.php		✓	302	502	HTML
23	http://localhost	GET	/portal.php			200	23702	HTML
24	http://localhost	GET	/logout.php			302	896	HTML
25	http://localhost	GET	/login.php			200	4363	HTML

Below the table, the 'Request' and 'Response' panes are expanded for the second row (HTTP/1.1 302 Found). The 'Request' pane shows the raw GET request to /logout.php. The 'Response' pane shows the raw HTTP response with status 302 Found, headers including Date, Server, X-Powered-By, and Expires, and a Location header pointing to /portal.php. To the right, the 'Inspector' pane shows 'Request Attributes' and 'Request Cookies'. A context menu is open over the response body, with the 'Send to Comparer' option highlighted.

Porównywanie realizowane jest na poziomie bajtów (pożerająca więcej zasobów komputerowych) lub tekstu (tokenizacja wyrazów oddzielonych spacją).

Możemy maksymalnie porównywać ze sobą dwa elementy jednocześnie. Opcja szczególnie przydatna, podczas analizy danych w których różnice są ciękkie do wyłapania.

The screenshot shows the 'Word compare' feature in Burp Suite. It compares two responses: one with length 896 (HTTP/1.1 302 Found) and one with length 502 (HTTP/1.1 302 Found). The left pane shows the response for length 896, which includes various HTTP headers and a Location: /portal.php. The right pane shows the response for length 502, which includes similar headers and a different Location: /portal.php. Below the panes, a 'Key' section indicates 'Modified', 'Deleted', and 'Added' differences. A 'Sync views' checkbox is at the bottom right.

Za przykład możemy dwie odpowiedzi, jedna po poprawnym logowaniu (`/login.php`), druga po poprawnym wylogowaniu (`/logout.php`). Jak widać zmieniają się tu tylko ciasteczka, więc można wywnioskować, że za to czy użytkownik jest zalogowany czy nie w aplikacji odpowiadają ciasteczka.

## Logger

Narzędzie do śledzenia aktywności sieciowej, która przechodzi przez burpa. Zostaje tu wyświetlony cały ruch sieciowy, nawet ten poza naszym zakresem (**scope**). Przydatne jeśli chcemy podejrzeć jak wyglądają żądania po naszej modyfikacji. Chociaż ta zakładka jest **read-only**, możemy wysłać żądania do innych narzędzi w obrębie programu. Tabela umożliwia nam sortowanie po wybranych kolumnach i filtrowanie.

Burp Suite Community Edition v2022.9.6 - Temporary Project

Capture filter: Logger memory limit set to 50MB | Capturing requests up to 1MB; capturing responses up to 1MB

View filter: Showing all items

#	Time	Tool	Method	Host	Path	Query	Param count	Status	Length	Start response timer
1	06:21:33 25 Jan 2023	Proxy	GET	localhost	/install.php		1	200	2483	3
2	06:21:36 25 Jan 2023	Proxy	GET	localhost	/install.php	install=yes	2	200	2487	22
3	06:22:08 25 Jan 2023	Proxy	GET	www.google.com	/search	q=bawapp&oq=bawapp&gs...	11	200	445903	149
4	06:22:07 25 Jan 2023	Proxy	GET	www.google.com	/imagesbranding/google...		6	200	4481	94
5	06:22:08 25 Jan 2023	Proxy	GET	www.google.com	/taipei.png		6	200	918	67
6	06:22:08 25 Jan 2023	Proxy	GET	www.google.com	/images/searchbox/desk...		6	200	1310	98
7	06:22:08 25 Jan 2023	Proxy	GET	www.google.com	/images/nav_logo321.we...		6	200	5839	86
8	06:22:08 25 Jan 2023	Proxy	GET	www.google.com	/gen_204	s=web&t=aft&typ=cis...	19	204	712	249
9	06:22:08 25 Jan 2023	Proxy	POST	www.google.com	/inouttools/images/ta...	ono	0	200	917	70
10	06:22:08 25 Jan 2023	Proxy	GET	www.ostatic.com						

Request

Pretty Raw Hex

Response

Pretty Raw Hex Render

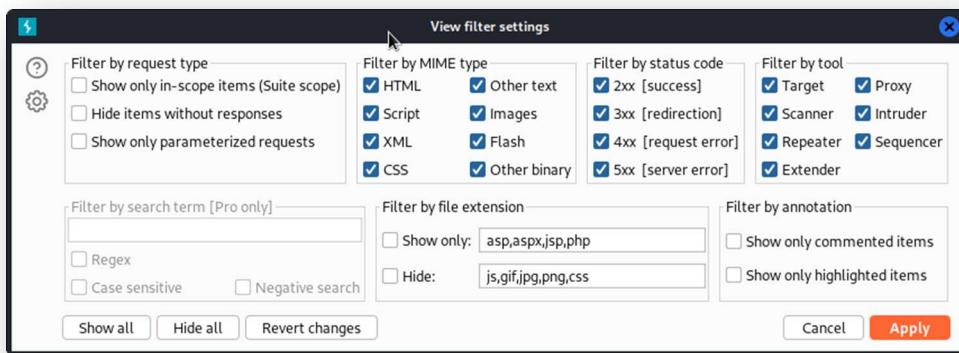
Inspector

Request Attributes

Request Cookies

Request Headers

Response Headers



Sam logger ma limit 50Mb po przekroczeniu którego zostaną usunięte najstarsze wpisy.

## Extender

Pozwala rozszerzyć funkcjonalność burpa poprzez instalację narzędzi utworzoną przez społeczność burpa.

Przykład: Szukanie podatności SQL Injection

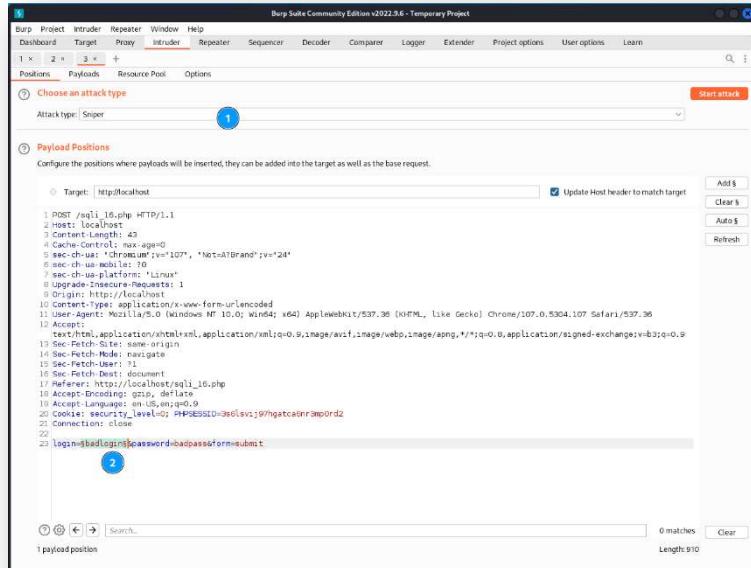
Udajemy się na podatną stronę zainstalowaną na naszej lokalnej maszynie, która działa pod adresem **localhost/sqli\_16.php**. Robimy to przez przeglądarkę skonfigurowaną przez burpa z poziomu

**Proxy > Intercept.** Wpisujemy błędne dane uwierzytelniania, których nie znamy. Na tym etapie ważne jest dla nas po prostu wykonanie żądania, które zostanie przepuszczone przez burpa.

Po naciśnięciu przycisku **Login** zostanie wygenerowane żądanie do serwera, który zweryfikuje czy podane przez login i hasło są prawidłowo. Jak już wiemy burp pośredniczy w komunikacji klient – serwer, dlatego też żądanie powinno być widoczne w historii **Proxy > HTTP history**.

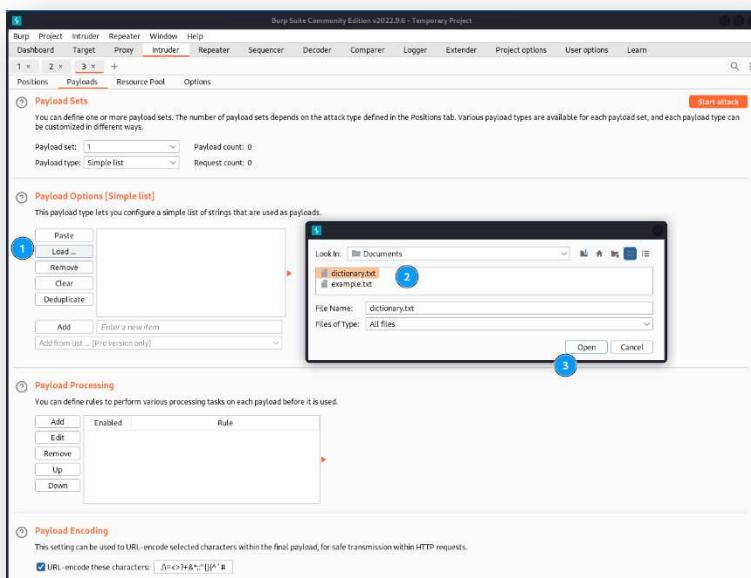
To samo mogliśmy wywnioskować z samej tabeli wyświetlającej historię żądań (1). Możemy zauważać żądanie w formie surowej z parametrami (2) oraz odpowiedź wyrenderowaną przez burpa, ponieważ użyliśmy opcji wyświetlania **Render** (3).

Żeby sprawdzić podatność SQLi chcemy sprawdzić czy wpisując składnię języka SQL do formularza aplikacja **backendowa** przepuści zainfekowany tekst do bazy danych. Jeśli tak, otrzymamy odpowiedź z zawartością bazy danych. Dostępnych jest wiele metod SQLi jak również wiele składni samego języka zapytań, dlatego warto było by posiadać już jakiś zbiór możliwych fragmentów składni SQL, który moglibyśmy sprawdzić. Możemy to zrobić pobierając z Internetu jakikolwiek gotowy słownik.



Wysyłamy żądanie do intrudera **PPM na żądanie > Send to intruder**. Jako metodę ataku wybieramy **Sniper (1)**, a za znacznik login. Metoda sniper podstawi za znacznik każdą wartość ze słownika w miejsce loginu i wyśle tak spreparowane żądanie.

Samo załadowanie słownika robimy z poziomu zakładki **Intruder > Payloads**



Rozpoczynamy atak przyciskiem **Start Attack**. Wersja community przedłuży nam cały proces, ale otrzymamy wyniki. Powinniśmy szukać wyników, które różnią się od reszty, zdecydowana większość ma rozmiar 13492, więc raczej nie weźmiemy ich pod uwagę. Mniejsze odpowiedzi natomiast po wyrenderowaniu dają nam błąd z bazy danych udowadnia istnienie podatności SQLi.

Request	Payload	Status	Error	Timeout	Length	Comment
148	and (select substring(@@versi...	200			2452	
149	and (select substring(@@versi...	200			2452	
150	and (select substring(@@versi...	200			2452	
151	and (select substring(@@versi...	200			2452	
152	and (select substring(@@versi...	200			2452	
153	and (select substring(@@versi...	200			2452	
27	AND 1=1 AND '%!='	200			2453	
28	AND 1=0 AND '%!='	200			2453	
142	RLIKE (SELECT CASE WHEN (4...	200			2457	
143	RLIKE (SELECT CASE WHEN (4...	200			2457	
31	AND 1083=1083 AND ('1427=1...	200			2459	
32	AND 7506=9091 AND ('5913=5...	200			2459	
33	AND 7300=7300 AND 'pkIZ=p...	200			2461	
34	AND 7300=7300 AND 'pkIZ=p...	200			2461	
35	AND 7300=7300 AND ('pkIZ=...	200			2461	
36	AND 7300=7300 AND ('pkIZ=...	200			2461	
13	OR 3409=3409 AND ('pytW' Li...	200			2466	
14	OR 3409=3409 AND ('pytW' Li...	200			2466	
0		200			13492	
1	OR 1=1	200			13492	
2	OR 1=0	200			13492	
3	OR x=x	200			13492	

## 4. NARZĘDZIA DO ŁAMANIA I TESTOWANIA HASEŁ

### 4.1. John The Ripper

John the Ripper to popularne narzędzie do łamania haseł w celu sprawdzenia podatności na ich złamanie. Jest to narzędzie typu open source, tj. kod źródłowy jest bezpłatnie udostępniany użytkownikom i może być modyfikowany oraz rozpowszechniany bez uiszczenia opłat. John the Ripper używany jest najczęściej do wykrywania słabych haseł, które mogą zagrozić bezpieczeństwu np. sieci lub innych podmiotów administracyjnych. Oprogramowanie może być używane w każdym systemie operacyjnym, lokalnie lub zdalnie za pomocą skryptów. Początkowo jednak opracowany został tylko dla systemów opartych na Uniksie, jednak teraz dostępny jest na około piętnastu różnych platformach (w tym Windows). Program łamie hashe za pomocą ataku słownika lub ataku siłowego. Formaty, które obsługuje to DES,RSA,MD4 i MD5, Kerberos AFS oraz hasze Windows LM.

Najważniejsze komendy

**john** – uruchamia program John the Ripper

**john --help** – wyświetla listę dostępnych opcji i parametrów

**john --test** – uruchamia testowanie szybkości i skuteczności programu

**john --wordlist=file.txt** – używa wskazanego pliku zawierającego listę słów jako słownika do ataku

**john --rules** – uruchamia atak z użyciem reguł permutacji haseł

**john --show** – wyświetla odgadnięte hasła

**john --incremental** – uruchamia atak inkrementalny

**john --session=sessionname** – umożliwia zapisanie i kontynuowanie sesji odgadywania haseł

Czym tak naprawdę jest hash?

Pojęcie to będzie bardzo często używane w dalszej części testowania oprogramowania. Jest to określony ciąg znaków podany przez użytkownika, który został przekształcony dzięki funkcji na krótką wartość znakową, posiadającą stały rozmiar. Właściwością hasza jest to, że jest on nieodwracalny.

Hash po zastosowaniu wybranej funkcji zawsze będzie taki sam tj. w przypadku jeżeli wykonamy daną funkcję na określonym ciągu znaków dowolną ilość razy, to wygenerowany hash (informacja wyjściowa) zawsze będzie taki sam.

Do najbardziej popularnych funkcji skrótu możemy zaliczyć:

- **MD5** (32 znaki / 128 bitów)
- **SHA-1** (40 znaków / 160 bitów)
- **SHA-256** (64 znaki / 256 bitów)

Hashe są wykorzystywane przede wszystkim dla bezpieczeństwa użytkowników i ich haseł. W przypadku wycieku bazy danych w której znajdują się hasła, hakerzy nie poznają hasła jakiego użyliśmy, a jedynie wartość hash, która została wcześniej wygenerowana.

### Dictionary attack (atak słownikowy)

Atak słownikowy jest jednym z rodzajów ataku na hasła, w którym atakujący używa listy słów znajdujących się w słowniku (lub listy haseł) do próby złamania hasła. Metoda ta polega na próbie zgadnięcia hasła, próbując różnych kombinacji słów z słownika.

W przypadku dictionary attack **John the Ripper** bierze każde słowo z słownika (dictionary) i próbuje użyć go jako hasła do odszyfrowania. Jeśli hasło pasuje, John the Ripper wyświetli je na ekranie jako hasło złamane.

Dictionary attack jest szybką i łatwą metodą ataku, ale jest również jedną z mniej skutecznych, ponieważ wiele haseł nie znajduje się w słowniku lub jest zmienione przez użytkowników (np. dodanie liczb lub znaków specjalnych).

### Brute-force attack (atak siłowy)

Atak siłowy (Brute-force attack) polega na próbie złamania hasła poprzez próbę wszystkich możliwych kombinacji znaków. Atakujący próbuje zgadnąć hasło poprzez próby wpisania każdej możliwej kombinacji liter, cyfr i znaków specjalnych.

Atak siłowy jest jednym z najmniej skutecznych sposobów na złamanie hasła, ale jest jednocześnie najbardziej skutecznym sposobem na złamanie silnego i skomplikowanego hasła. W przypadku silnych haseł, które składają się z wielu znaków, atak siłowy może trwać bardzo długo, a nawet być niemożliwy do zakończenia.

Aby zwiększyć skuteczność ataku siłowego, można użyć różnych taktyk, takich jak maskowanie, które pozwala na ograniczenie liczby prób, lub użyć wielu procesorów lub GPU.

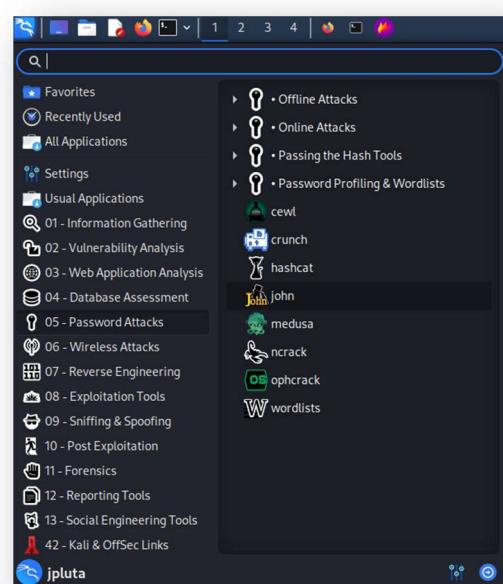
Należy jednak pamiętać, że atak siłowy jest nieetyczny i niezgodny z prawem, nie należy stosować tej metody bez pozwolenia odpowiednich podmiotów.

### Uruchomienie

Aby zacząć korzystać z John'a wystarczy w systemie Kali Linux wybrać z menu pozycję:

**Aplikacje → Password Attacks → john.**

Po uruchomieniu pokaże nam się wersja, z której aktualnie korzystamy.

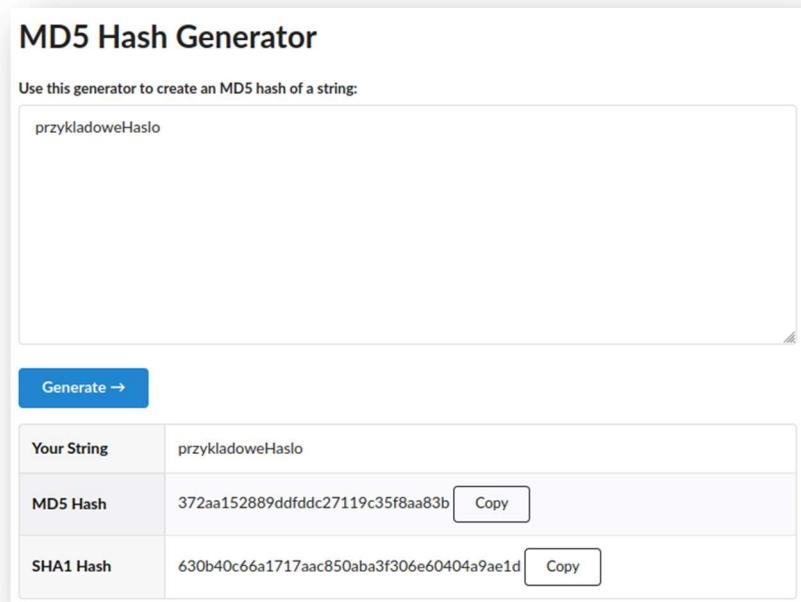




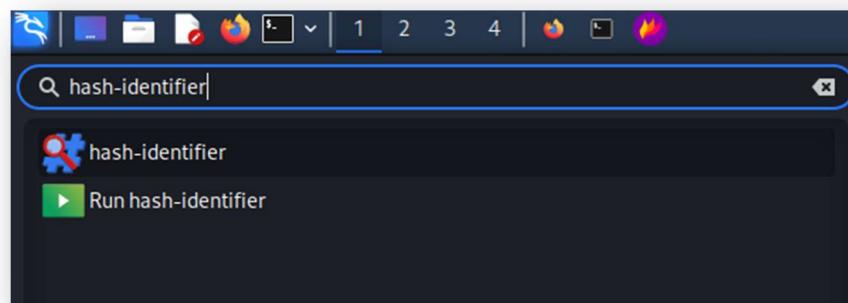
```
jpluta@kali: ~
File Actions Edit View Help
$ john
John the Ripper 1.9.0-jumbo-1+bleeding-aec1328d6c 2021-11-02 10:45:52 +0100 OMP [linux-gnu 64-bit x86_64 AVX2 AC]
Copyright (c) 1996-2021 by Solar Designer and others
Homepage: https://www.openwall.com/john/
Usage: john [OPTIONS] [PASSWORD-FILES]
Use --help to list all available options.
[jpluta@kali: ~]
```

### Atakowanie haseł formatu MD5 – dictionary attack

Najpierw musimy przygotować listę znaków (lub przynajmniej jedno hasło jak w naszym przypadku). Zaczniemy od wygenerowania hasza MD5 (32 znakowy; 128 bitowy) wybranego przez nas hasła. Aby to zrobić, należy wejść na przykładową stronę, np. [md5hashgenerator.com](https://md5hashgenerator.com).



Na początku powinniśmy zidentyfikować typ używanego hasza dzięki programowi **hash-identifier**; możemy go znaleźć w wyszukiwarce w naszym systemie.



Kopiujemy hash ze strony, w którym go wygenerowaliśmy i wklejamy do konsoli. Jak widzimy, program mówi nam, że możliwym hashem jest docelowy MD5, którego chcemy użyć.

Następnym krokiem będzie skopiowanie i zapisanie hasza do pliku (przykładowo **haslo.txt**).

```
jpluta@kali: ~/Desktop
File Actions Edit View Help
└── (jpluta@kali)-[~]
$ cd Desktop
└── (jpluta@kali)-[~/Desktop]
$ sudo nano haslo.txt
File System
```

Teraz poprzez komendę:

```
sudo john --format=RAW-MD5 haslo.txt
```

```
(jpluta㉿kali)-[~/Desktop]
$ sudo john --format=RAW-MD5 haslo.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=3
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
Og 0:00:02:17 3/3 0g/s 30396Kp/s 30396Kc/s 30396KC/s ms.ww64c..ms.wning
```

spróbujemy złamać nasze hasło. Jak widzimy, tego typu problem przedstawiony na zdjęciu oznacza, że hasło nie znajduje się na liście słów w pliku **password.lst** (jest to domyślny plik listingu, na którym opiera się program).

```
(jpluta㉿kali)-[~/Desktop]
$ sudo john --format=RAW-MD5 haslo.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=3
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
Proceeding with incremental:ASCII
0g 0:00:02:17 3/3 0g/s 30396Kp/s 30396Kc/s ms.ww64c..ms.wning
```

```
(jpluta㉿kali)-[~/Desktop]
$ sudo cat /usr/share/john/password.lst
```

Kopiujemy ścieżkę zaznaczoną poniżej i próbujemy ją otworzyć za pomocą edytora tekstu.

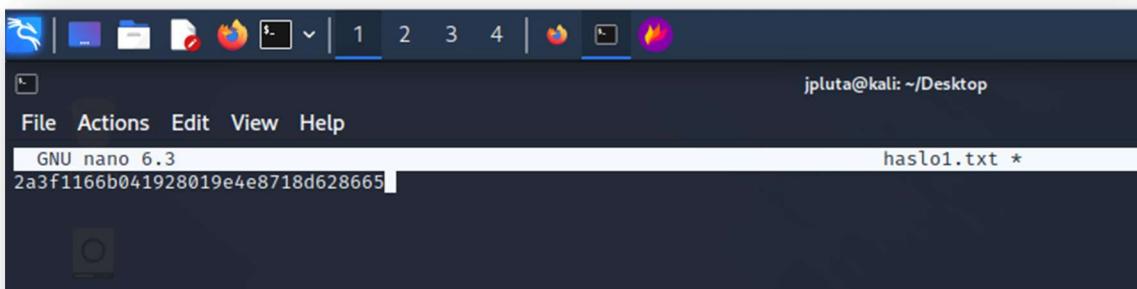
Lista haseł jest bardzo długa, ale niestety nie znajduje się w niej nasze przykładowe hasło, które zapisaliśmy w pliku **haslo.txt**. (na zrzucie ekranu widać tylko przykładowe pozycje).

```
File Actions Edit View Help
jpluta@kali: ~/Desktop
visla
www
y
zzz
1337
1950
3141
3533
4355
4854
6301
Bonzo
ChangeMe
Front242
Gretel
Micheli
Noriko
Sidekick
Sverige
Swoosh
Woodrow
aa
ayelet
barn
betacam
bla
bohat
cuda
doc
ha
hallowell
haro
hosehead
i
ilmari
imeli
jillz3
jel
kcln
kerry
kerry2
leaf
lisabon
mart
matt11
mech
moncats
panogl
performa
prof
ratio
shop
slip
stivers
tapani
tergas
terz
test3
tula
unix
uxv1
xanth
!@#$%^&*
17old
333376
Qwert
allo
dirk
go
newcourt
nite
notused
sss
```

A red box highlights the following words:

Bonzo  
ChangeMe  
Front242  
Gretel  
Micheli  
Noriko  
Sidekick  
Sverige  
Swoosh  
Woodrow  
aa  
ayelet  
barn  
betacam

W takim razie musimy dodać nasze hasło do tej listy słów, lub spróbować z innym, znajdującym się już tutaj hasłem. Wybieramy przykładowe hasło (np. *ship*) i powtarzamy procedurę z wygenerowaniem hasha MD5 na stronie. Tworzymy następnie w taki sam sposób plik, w którym będzie znajdować się nasz hash, nazwijmy go przykładowo **haslo1.txt**.

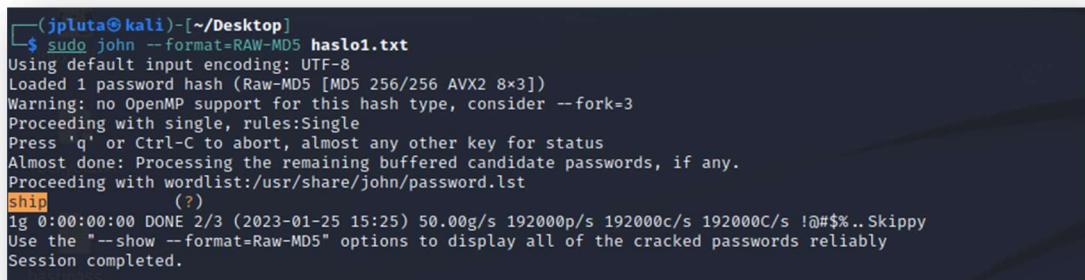


A screenshot of a terminal window titled "haslo1.txt". The window shows the command "GNU nano 6.3" at the top, followed by the content of the file: "2a3f1166b041928019e4e8718d628665". The terminal interface includes a menu bar with "File", "Actions", "Edit", "View", and "Help", and a status bar indicating the user is at the root prompt "jpluta@kali: ~/Desktop".

Próbowejmy tej samej metody komendą:

```
sudo john --format=RAW-MD5 haslo1.txt
```

Ważne jest, abyśmy znajdowali się aktualnie w miejscu, gdzie mamy zapisany plik tekstowy z haszem (w moim przypadku jest to pulpit, więc w terminalu wpisujemy **cd ~/Desktop** i następnie komendę podaną powyżej). Jak widzimy poniżej, udało się złamać hasło.



```
(jpluta@kali)-[~/Desktop]
$ sudo john --format=RAW-MD5 haslo1.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=3
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
ship      (?)
1g 0:00:00:00 DONE 2/3 (2023-01-25 15:25) 50.00g/s 192000p/s 192000c/s 192000C/s !@#$%..Skippy
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.
```

## Atakowanie haseł formatu MD5 – brute-force

Tak samo jak w poprzednim przykładzie musimy najpierw przygotować listę znaków. Podobnie skorzystamy z generatora na stronie [md5hashgenerator.com](http://md5hashgenerator.com).

MD5 Hash Generator

Use this generator to create an MD5 hash of a string:

Generate →

Your String	kali
MD5 Hash	d6ca3fd0c3a3b462ff2b83436dda495e <button>Copy</button>
SHA1 Hash	e7e971e55af10f713238780785ec5e63720509f0 <button>Copy</button>

Jako przykładowe hasło będziemy używać słowa „*kali*”. Nie jest to najczęstsze do złamania hasło i raczej nie powinniśmy go używać w codziennym życiu do zabezpieczania naszych kont, jednak w tym przypadku chcemy tylko pokazać możliwości programu i nie chcemy aby program działał bardzo długo. Analogicznie jak dla ataku słownikowego tworzymy plik tekstowy z hashem MD5.

Pora na uruchomienie ataku. Wpisujemy komendę:

```
john --incremental --format=raw-md5 hasloBrutalForce.txt
```

W tym przypadku John będzie używał metody **incremental** (stopniowego zwiększania długości hasła) i formatu raw-md5 do próby złamania hasła „*kali*” z pliku **hasloBrutalForce.txt**. Jak widać zajęło to programowi niecałą sekundę, ponieważ hasło było bardzo proste.

```
(jpluta㉿kali)-[~]
$ john --incremental --format=raw-md5 hasloBrutalForce.txt
Using default input encoding: UTF-8
Loaded 1 password hash (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=3
Press 'q' or Ctrl-C to abort, almost any other key for status
kali (?)
1g 0:00:00:01 DONE (2023-01-25 17:16) 0.6944g/s 1798Kp/s 1798Kc/s 1798KC/s kieu..kyot
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

(jpluta㉿kali)-[~]
$
```

## Maskowanie

(ang. **masking**) Jest to taktyka stosowana podczas ataku siłowego, która polega na ograniczeniu liczby prób złamania hasła poprzez określenie maski hasła. Maska hasła to ciąg znaków, który określa jakie znaki mogą wystąpić w danych pozycjach hasła. John the Ripper będzie próbował tylko takich kombinacji znaków, które pasują do tej maski, co znacznie zwiększa szybkość ataku.

Spójrzmy teraz na parę przykładów:

```
└─(jpluta㉿kali)-[~]
$ john --incremental=All --format=raw-md5 --mask='?l?l?l?l?l' hashfile.txt
```

atak siłowy z ograniczeniem do tylko małych liter

```
└─(jpluta㉿kali)-[~]
$ john --incremental=All --format=raw-md5 --mask='?u?u?u?d?d?d' hashfile.txt
```

atak siłowy z ograniczeniem do pierwszych trzech znaków duże litery i pozostałe

```
└─(jpluta㉿kali)-[~]
$ john --incremental=All --format=raw-md5 --mask='?l?l?l?l?d?d' hashfile.txt
```

atak siłowy z ograniczeniem do pierwszych czterech znaków małe litery i pozostałe

**Masking** pozwala na skupienie sił na prawdopodobniejszych kombinacjach znaków, co zwiększa skuteczność ataku siłowego. Warto jednak pamiętać, że im bardziej zawężona jest maska, tym dłużej będzie trwał atak.

Atakowanie haseł z plików jpg, pdf – dictionary attack

John the Ripper może być użyty do łamania haseł z plików, takich jak pliki JPG, PDF itp. Oznacza to, że John the Ripper może próbować odkodować hasła zabezpieczające te pliki, aby uzyskać dostęp do zawartości.

```
└─(jpluta㉿kali)-[~]
$ john --format=pdf --wordlist=dictionary.txt encrypted.pdf
```

W tym przykładzie, John the Ripper używa formatu PDF, słownika o nazwie **dictionary.txt** i pliku pdf o nazwie **encrypted.pdf**. John the Ripper próbuje złamać hasło za pomocą słów z słownika, a jeśli znajdzie prawidłowe hasło, zostanie ono wyświetcone na ekranie.

## Atakowanie plików zip – dictionary attack

Programu możemy użyć także do odgadnięcia hasła, które zabezpiecza dostęp do pliku ZIP. Aby przeprowadzić atak na plik **ZIP** należy jak w poprzednich przykładach utworzyć lub wykorzystać już istniejący plik słownika z potencjalnymi hasłami.

```
(jpluta㉿kali)-[~]
$ john --format=zip --wordlist=dictionary.txt file.zip
```

W tym przykładzie używamy pliku słownika **dictionary.txt** do próby złamania hasła pliku **file.zip** i formatu ZIP. John the Ripper przeszuka plik słownika i spróbuje znaleźć hasło, które pozwoli na otwarcie pliku. Jeśli znajdzie hasło, zostanie wyświetcone na ekranie.

## 5. NARZĘDZIA DO ZABEZPIECZANIA POŁĄCZEŃ SIECIOWYCH

### 5.1. Nmap

**Nmap** (Network Mapper) jest programem do skanowania i odkrywania konfiguracji sieci oraz do audytu bezpieczeństwa. Narzędzie działa z poziomu konsoli poleceń, posiada wiele przełączników, które dostarczają wiele opcji skanowania. Nmap jest używany do:

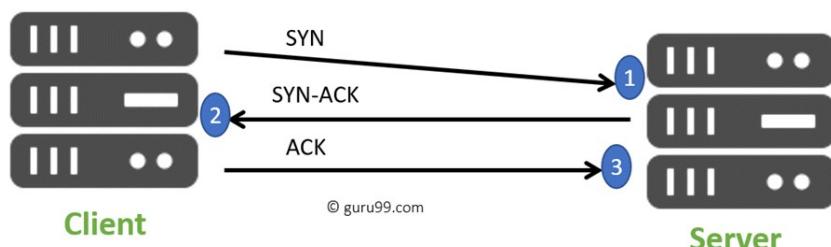
- Odkrywanie otwartych portów i usług sieciowych
- Odczytywanie wersji, konfiguracji i właściwości usług sieciowych
- Odkrywanie systemu operacyjnego na maszynie docelowej
- Odnajdywanie dokładnej trasy do hosta docelowego
- Monitorowanie hostów

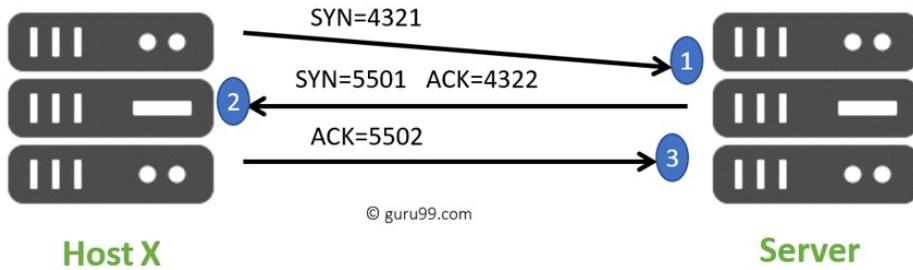
Skanowanie sieci na bazie protokołu TCP

Skan TCP

Podstawą komunikacji jest tu protokół TCP (Transfer Control Protocol), który bazuje na trójstopniowym uzgadnianiu połączenia (**three-way handshake**) pomiędzy maszyną, z której przeprowadzany zostaje skan a hostem docelowym. Z reguły komunikacja za pomocą TCP jest łatwa do wykrycia i niepraktyczna w przypadku potrzeby zestawienia „cichego” połączenia. W praktyce schemat wysłania **pakietu** za pomocą TCP wygląda następująco:

1. Host źródłowy wysyła sygnał **SYN (Synchronization)** do hosta docelowego w celu nawiązania połączenia i oznajmia, że jest gotowy do komunikacji z nim – dokonuje synchronizacji. Dołącza losowy, 4-bajtowy numer sekwencji, który będzie kluczowy w przesyłaniu danych, maksymalny rozmiar segmentu TCP oraz wielkość okna i ewentualny jego mnożnik.
2. Host docelowy odpowiada sygnałem **SYN-ACK (Synchronization-Acknowledgment)**, zaznacza potwierdzenie otrzymania wiadomości oznaczając bajt w polu ACK na sekwencję przeslaną ze źródła powiększoną o 1 – zgłasza, że po przyjęciu pakietu od hosta źródłowego jest gotowy na komunikację. Dalsze wartości wpisywane w pole ACK będą rzeczywistymi danymi, które zostaną przesłane przez TCP z drugiej strony komunikacji, tylko w tym przypadku jest tworzony „**ghost byte**” lub „**phantom byte**”, aby mogło dojść do komunikacji. Host docelowy rozpoczyna swoją synchronizację w celu wymiany danych z hostem źródłowym, tworzy swoją wartość sekwencji.
3. Host źródłowy wysyła sygnał **ACK (Acknowledgment)**, do pola potwierdzenia zostaje wpisany numer sekwencji z hosta wysyłającego sygnał SYN-ACK. Teraz oba urządzenia są gotowe do przeprowadzenia komunikacji między sobą i będą to robić przez TCP wysyłając sygnały ACK z określonymi danymi, których parametry są nadane w nagłówkach TCP.





### Skan UDP

przeznaczony do nasłuchiwanego przychodzących żądań w protokole **UDP** na otwartych portach. UDP, w porównaniu do TCP nie ma żadnego mechanizmu, który potwierdziłby uczestnictwo obu stron w komunikacji, dlatego zawsze jest możliwość odebrania fałszywego sygnału (**false-positive**). Jednakże takie skanowanie ma swoje zastosowania w odkrywaniu szkodliwego oprogramowania typu trojan. W przypadku wysyłania i odbierania sygnałów UDP trzeba liczyć, że obciążone pasmo będzie zwalniało transfer, toteż skanowanie może okazać się nieefektywne.

### Skan SYN

wykorzystuje sygnał synchronizujący z protokołem TCP, jednakże po otrzymaniu od maszyny docelowej pakietu z sygnałem SYN-ACK połączenie nie zostaje uformowane, ponieważ w drugą stronę nie zostaje przekazana wiadomość potwierdzenia (ACK). Ma to swoje zalety w postaci mniej wykrywalnego skanu, co może pomóc dokonać taką operację „po cichu” (**stealthy**). Nmap jest w stanie zebrać potrzebne informacje tylko na podstawie sygnału synchronizującego.

### Skan ACK

służy do sprawdzenia stanu portów, może powiedzieć czy dany port posiada jakiekolwiek filtrowanie, np. w postaci **firewalla**. Zachodzące połączenie w przypadku wysłania sygnału ACK jednoznacznie określa parametry, na podstawie których pakiety są filtrowane w maszynie docelowej.

### Skan FIN

również „cichy” rodzaj nawiązywania połączenia, działa podobnie jak SYN, z tą różnicą, że jest to po prostu pakiet, który mówi o zakończeniu połączenia. Urządzenia sieciowe po otrzymaniu sygnału FIN najczęściej odsyłają pakiet RST, do resetowania połączenia. Często ten typ ataku jest cięższy do wykrycia dla programów zajmujących się ochroną przed atakami.

### Inne typy skanowania sieci

Innymi skanami, które ciężko wykryć są np.: skan typu **NULL**, który wysyła ramkę, gdzie wszystkie parametry są ustawione na wartość *null*, co jest ciężkie do weryfikacji dla urządzeń sieciowych, ponieważ pakiet z takimi wartościami nie ma swojej fizycznej reprezentacji, toteż nie ma określonych reguł, co robić kiedy taki sygnał przyjdzie; skan typu **XMAS**, który również operuje na wartościach parametrów w nagłówku TCP; skan **RPC** wysyła sygnał do maszyn i te które obsługują usługi Remote Procedure Call, są podatne na przekazanie informacji tych zdalnych usługach; skan typu **IDLE** – najmniej wykrywalny ze wszystkich podatnych, jego wykorzystywanie podchodzi pod złośliwy atak, także nie będzie tutaj przedstawiany rezultat jego działania.

## Komendy do skanowania sieci

Teraz przypatrzmy się jak wykonać skanowanie sieci używając programu Nmap. Program ten używa terminala do wykonania czynności związanych ze skanowaniem, dokonuje listingu odnalezionej adresów, portów i usług. W tabeli przedstawiono niektóre z możliwych konfiguracji

Przełącznik	Typ skanu	Przykładowa komenda
-sS	TCP SYN skan portów (stealthy)	nmap -sS 192.168.0.1
-sT	TCP skan portów	nmap -sT 192.168.0.1
-sU	UDP skan portów	nmap -sU 192.168.0.1
-sA	TCP ACK skan portów	nmap -sA 192.168.0.1

### Skanowanie sieci po „cichu”

Aby dokonać cichego skanu użyjemy przełącznika **-sS**, który nie dokona potwierdzenia połączenia, toteż będzie cięższy w wykryciu. Operację przeprowadziłem z maszyny wirtualnej na moj adres hosta (w tym przypadku Windows) poprzez zmostkowanie adresów w ustawieniach VirtualBox'a. Tak wygląda rezultat wykonania komendy:

```
(root㉿kali)-[~/home/kali]
└─# nmap -sS 192.168.0.158
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 15:09 EST
Nmap scan report for 192.168.0.158
Host is up (0.0022s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
3580/tcp   open  nati-srvloc

Nmap done: 1 IP address (1 host up) scanned in 5.08 seconds
```

Taki eksperyment trwał nieco ponad 5 sekund i ujawnił nam już bardzo dużo cennych informacji, np. wypisał otwarte porty TCP.

Teraz spróbujemy zeskanować całą podsieć zmieniając adres hosta na adres sieci, w tym przypadku **192.168.0.1** i dodając na koniec **suffix maski sieciowej /24**

```
(root㉿kali)-[~/home/kali]
└─# nmap -sS 192.168.0.1/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 16:44 EST
Stats: 0:00:04 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 0.05% done
Stats: 0:00:07 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 1.36% done; ETC: 16:51 (0:06:03 remaining)
Stats: 0:00:10 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.49% done; ETC: 16:50 (0:05:14 remaining)
Stats: 0:00:20 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.50% done; ETC: 16:56 (0:11:04 remaining)
Stats: 0:00:20 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 2.51% done; ETC: 16:56 (0:11:01 remaining)
```

Jak można zauważyć, taka operacja została oszacowana na 11 minut pracy programu. Jest to spowodowane tym, że każdy z 254 hostów w podsieci musi otrzymać pakiet i odesłać swój z powrotem. Takie skanowanie oczywiście może się powieść, ale wymaga ono cierpliwości. Nmap przychodzi jednak z pewnym usprawnieniem i udostępnia przełącznik **-F**, który dokonuje szybkiego skanu.

Przykład użycia takiej komendy przedstawiono poniżej:

```
[root@kali]# nmap -F -sS 192.168.0.1/24
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 16:50 EST
Stats: 0:00:01 elapsed; 0 hosts completed (0 up), 256 undergoing Ping Scan
Ping Scan Timing: About 15.67% done; ETC: 16:50 (0:00:05 remaining)
Stats: 0:00:03 elapsed; 0 hosts completed (0 up), 256 undergoing Ping Scan
Ping Scan Timing: About 23.10% done; ETC: 16:50 (0:00:10 remaining)
Stats: 0:00:06 elapsed; 0 hosts completed (0 up), 256 undergoing Ping Scan
Parallel DNS resolution for 256 hosts. Timing: About 88.28% done; ETC: 16:50 (0:00:00 remaining)
Stats: 0:00:09 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 1.10% done; ETC: 16:54 (0:04:29 remaining)
Stats: 0:00:11 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 10.19% done; ETC: 16:51 (0:00:44 remaining)
Stats: 0:00:12 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 17.62% done; ETC: 16:51 (0:00:28 remaining)
Stats: 0:00:12 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 22.05% done; ETC: 16:50 (0:00:21 remaining)
Stats: 0:00:13 elapsed; 0 hosts completed (64 up), 64 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 22.92% done; ETC: 16:50 (0:00:24 remaining)
```

Została zwiększa szybkość wykonywania skanów, komunikacja z hostami odbywa się tutaj partiami, więc po kilku minutach program zwróci wynik dla pierwszych 64 hostów z podsieci.

Przykłady skanowania z innymi przełącznikami

Poniżej podano kolejne funkcje programu nmap:

Przełącznik	Typ skanu	Przykładowa komenda
-Pn	Tylko skan portów	nmap -Pn 192.168.0.1
-sn	Tylko skan hosta	nmap -sn 192.168.0.1
-PR	Odkrycie adresów przez protokół ARP w sieci lokalnej	nmap -PR 192.168.0.1
-n	Wyłączenie translacji DNS	nmap -n 192.168.0.1

Prosta komenda na odkrycie czy host jest aktywny:

```
[root@kali]# nmap -sn 192.168.0.158
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 17:00 EST
Nmap scan report for 192.168.0.158
Host is up (0.0033s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
```

Dla tych komend możemy również wyspecyfikować porty poprzez przełącznik:

**-p [nr\_portu][nr\_portu\_low-nr\_portu\_hi]**

W ten sposób możemy sprawdzać tylko konkretne porty na danym hoście. Możliwe jest podanie zakresu portów, zakresu adresów albo całej podsieci lub jednego hosta, także nmap oferuje wiele możliwości w tym zakresie.

```
(root㉿kali)-[~/home/kali]
# nmap -p 80 192.168.0.1
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 17:05 EST
Nmap scan report for 192.168.0.1
Host is up (0.0011s latency).

PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.18 seconds
```

W tym rezultacie widać że port 80, który odpowiada za protokół HTTP jest otwarty dla adresu sieci. To dlatego, że **ISP** zazwyczaj zamieszcza tam panel administracyjny z dostępem z przeglądarki.

#### Wykrywanie systemu operacyjnego hosta oraz wersji usług

Nmap umożliwia przeszukiwanie umożliwiające odkrycie systemu operacyjnego, na którym pracuje dany host, dodatkowo jest możliwe wyświetlenie usług jakie ma zainstalowane na swoim systemie, co jest aktualnie aktywne, z czego korzysta. W tabeli pokazano kolejne przełączniki komend:

Przełącznik	Typ skanu	Przykładowa komenda
-sV	<b>Wykrywanie wersji aktywnych usług</b>	<b>nmap -sV 192.168.0.1</b>
-A	<b>Agresywne skanowanie</b>	<b>nmap -A 192.168.0.1</b>
-O	<b>Wykrywanie systemu operacyjnego hosta docelowego</b>	<b>nmap -O 192.168.0.1</b>

#### Test skanowania agresywnego

```
(root㉿kali)-[~/home/kali]
# nmap -A 192.168.0.158
Starting Nmap 7.93 ( https://nmap.org ) at 2023-01-25 17:21 EST
Nmap scan report for 192.168.0.158
Host is up (0.00027s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT      STATE SERVICE          VERSION
135/tcp    open  msrpc          Microsoft Windows RPC
139/tcp    open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds   Microsoft Windows
3580/tcp   open  http           National Instruments LabVIEW service locator httpd 1.0.0
|_http-title: Site doesn't have a title (text/html).
|_http-server-header: NI Service Locator/1.0.0 (SLServer)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: bridge/general purpose
Running (JUST GUESSING): Oracle Virtualbox (98%), QEMU (92%)
OS CPE: cpe:/o:oracle:virtualbox cpe:/a:qemu:qemu
Aggressive OS guesses: Oracle Virtualbox (98%), QEMU user mode network gateway (92%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   311:
|     Message signing enabled but not required
| smb2-time:
|   date: 2023-01-25T22:21:52
|   start_date: N/A
|   clock-skew: 4s

TRACEROUTE (using port 80/tcp)
HOP RTT      ADDRESS
1  0.17 ms  10.0.2.2
2  0.14 ms  192.168.0.158

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.06 seconds
```

Taka operacja wykorzystuje wszystkie możliwości nmapa i pozyskuje jak najwięcej informacji o hoście. Są to między innymi: system operacyjny, uruchomione usługi, otwarte porty włącznie z usługami, trasa (**traceroute**) oraz wiele innych. Wykonanie takiego skanowania w cudzych sieciach może się okazać bardzo złośliwe, może ułatwić obcemu użytkownikowi włamanie. Dlatego trzeba zabezpieczać swoje sieci oraz należy korzystać z tych narzędzi tylko w izolowanych środowiskach bądź posiadając kwalifikacje do wykonania takich operacji w celach poprawy cyberbezpieczeństwa.

## 6. NARZĘDZIA DO PRZEPROWADZANIA AUDYTÓW BEZPIECZEŃSTWA

### 6.1. Lynis

Lynis to narzędzie konsolowe do wykrywania luk w zabezpieczeniach serwerów oraz monitorowania ich aktualnego stanu zabezpieczeń. Narzędzie to można również wykorzystać do audytowania dowolnej maszyny, którą ten program wspiera.

Lista przykładowych systemów Unix-based, które są wspierane przez to narzędzie:

- AIX
- FreeBSD
- HP-UX
- Linux
- macOS
- NetBSD
- NixOS
- OpenBSD
- Solaris

Środowisko pracy z oprogramowaniem

Testowanym systemem, który został wykorzystany, jest Kali GNU/Linux Rolling x86\_64 uruchomiony za pomocą wirtualizacji maszyny programem Oracle VM VirtualBox w wersji 7.0.6.

Specyfikacja i podstawowe informacje o środowisku testowym wyświetcone programem neofetch:

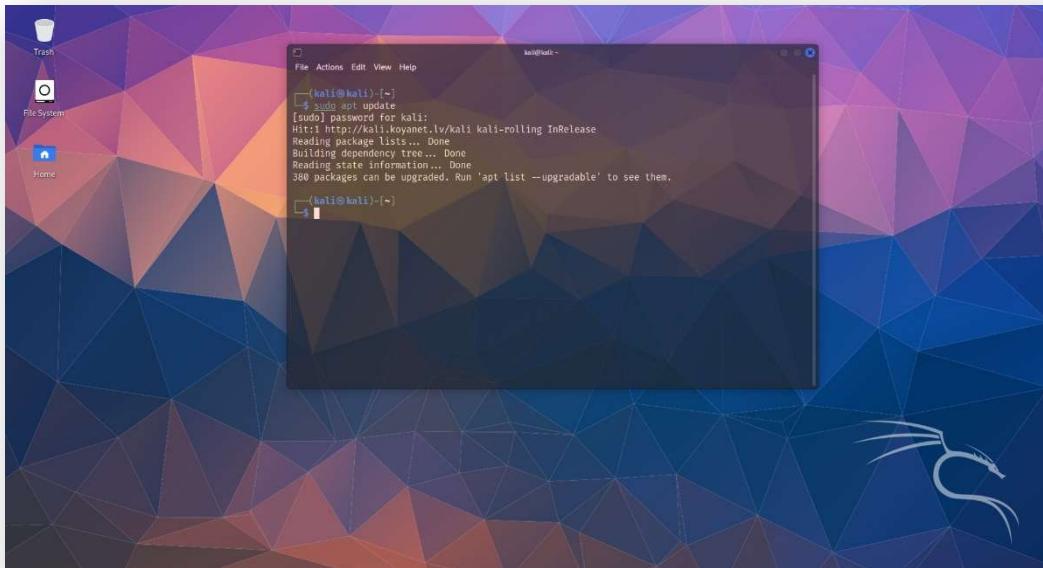
The screenshot shows a terminal window titled 'kali@kali: ~'. The user has run the 'neofetch' command. The output is split into two main sections. On the left, there is a stylized ASCII art representation of a person's head and shoulders. On the right, the system information is listed in a structured format:

```
kali@kali: ~
$ neofetch
.....(kali㉿kali)-[~]
.....$ neofetch
.....,:::ccc,.
.....''';lx0.
.....'',.,,:ld;
.....';:::;,.,x,
.....,0xoc:., ...
.....,ONkc:;,cokOdc',.
.....OMo          ':ddo.
.....dMc          ':00;
.....OM.          .:o.
.....;Wd          .:.
.....;XO,
.....,d00dlc;...
.....',;:cd00d:.,.
.....,.;d,';;
.....'d, .
.....;l ..
.....'.o
.....'c
.....'.'
```

kali@kali	
OS:	Kali GNU/Linux Rolling x86_64
Host:	VirtualBox 1.2
Kernel:	6.0.0-kali6-amd64
Uptime:	21 mins
Packages:	2631 (dpkg)
Shell:	zsh 5.9
Resolution:	1920x1080
DE:	Xfce 4.18
WM:	Xfwm4
WM Theme:	Kali-Dark
Theme:	Kali-Dark [GTK2], adw-gtk3-dark [GTK3]
Icons:	Flat-Remix-Blue-Dark [GTK3]
Terminal:	qterminal
Terminal Font:	Fira Code 13
CPU:	Intel i7-10750H (2) @ 2.592GHz
GPU:	00:02.0 VMware SVGA II Adapter
Memory:	698MiB / 3922MiB

**Uwaga:** System operacyjny celowo nie został zaktualizowany by wszelkie możliwe luki zabezpieczeń mogły być z większym prawdopodobieństwem wykryte przez narzędzie.

Po aktualizacji listy stanów repozytoriów, menedżer pakietów wykrył 380 możliwych aktualizacji:



## Instalacja

Poprzez wcześniej użyty manager pakietów/ system zarządzania pakietami APT zainstalowano Lynis. W konsoli należy wpisać komendę:

```
sudo apt install lynis
```

Wyświetlenie zainstalowanego pakietu ( wersja 3.0.8):

```
(kali㉿kali)-[~]
$ apt list | grep lynis
WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

lynis/kali-rolling,now 3.0.8-1.1 all [installed]

(kali㉿kali)-[~]
$
```

```
(kali㉿kali)-[~]
$ sudo lynis --version
[sudo] password for kali:
3.0.8
```

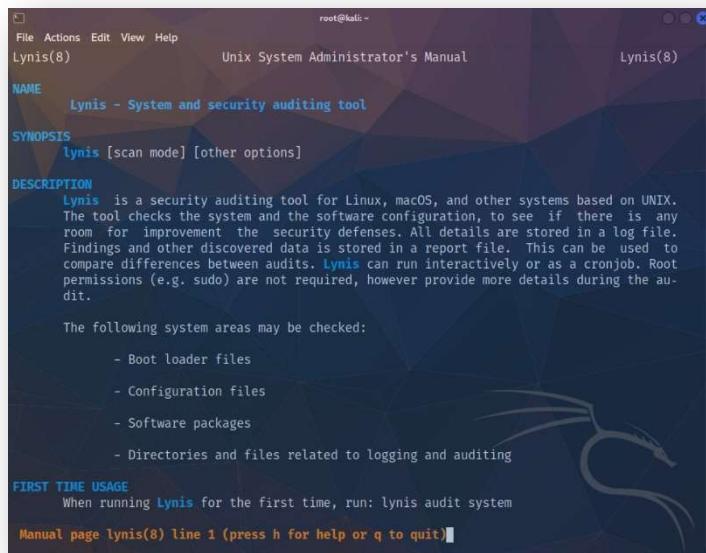
Program do działania nie wymaga podwyższonych uprawnień jednak takie mogą być potrzebne do raportowania szczegółów z całego systemu.

Obszary działania narzędzia oraz pierwsze uruchomienie  
Lynis opiera działanie na podanych niżej obszarach systemu:

- pliki programu rozruchowego
- pliki konfiguracyjne
- zainstalowane w systemie oprogramowanie
- katalogi i pliki skorelowane z logowaniem i autoryzacją

Jak można wyczytać z podręcznika do programu (komenda w konsoli **man lynis**), pierwsze uruchomienie programu uzyskujemy poprzez wprowadzenie w konsoli komendy:

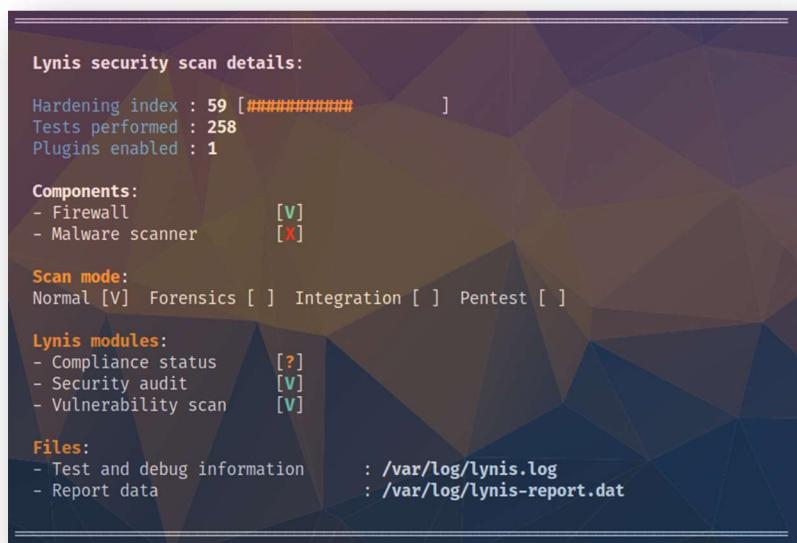
**lynis audit system**



The screenshot shows a terminal window with the Lynis manual page displayed. The title bar reads "rsst@kali: ~ Lynis(8) Unix System Administrator's Manual Lynis(8)". The content of the manual page includes:

- NAME**: Lynis - System and security auditing tool
- SYNOPSIS**: lynis [scan mode] [other options]
- DESCRIPTION**: Lynis is a security auditing tool for Linux, macOS, and other systems based on UNIX. The tool checks the system and the software configuration, to see if there is any room for improvement the security defenses. All details are stored in a log file. Findings and other discovered data is stored in a report file. This can be used to compare differences between audits. Lynis can run interactively or as a cronjob. Root permissions (e.g. sudo) are not required, however provide more details during the audit.
- The following system areas may be checked:
  - Boot loader files
  - Configuration files
  - Software packages
  - Directories and files related to logging and auditing
- FIRST TIME USAGE**: When running Lynis for the first time, run: lynis audit system
- Manual page lynis(8) line 1 (press h for help or q to quit)

Efekt pracy – raport ogólny



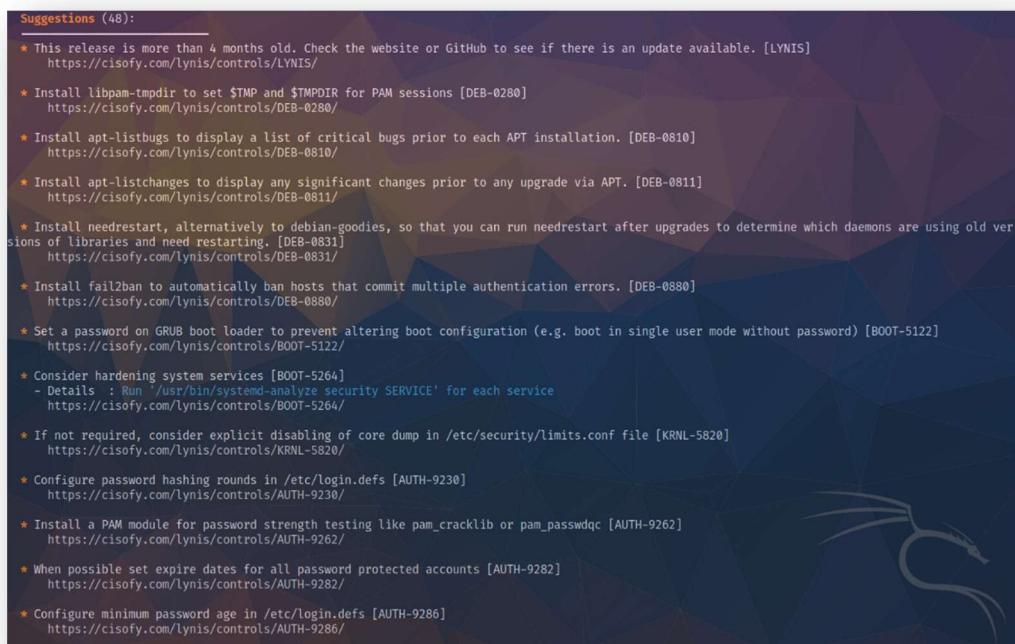
The screenshot shows the "Lynis security scan details" report. It displays the following information:

- Lynis security scan details:**
- Hardening index : 59 [#####]
- Tests performed : 258
- Plugins enabled : 1
- Components:**
  - Firewall [V]
  - Malware scanner [X]
- Scan mode:** Normal [V] Forensics [ ] Integration [ ] Pentest [ ]
- Lynis modules:**
  - Compliance status [?]
  - Security audit [V]
  - Vulnerability scan [V]
- Files:**
  - Test and debug information : /var/log/lynis.log
  - Report data : /var/log/lynis-report.dat

Po wykonaniu diagnozy systemu można odczytać podsumowanie wyświetcone w konsoli. Umieszczony jest wynik opisujący poziom zabezpieczeń (**59/100** na maszynie testowej) co jest wynikiem stosunkowo dobrym biorąc pod uwagę fakt, że nie wykonano żadnej interwencji w stan bezpieczeństwa systemu. Wyświetlana jest m. in. też informacja o ilości przeprowadzonych testów, rodzaju przeprowadzonego skanowania oraz sugestii uzupełnienia brakujących komponentów. Program generuje również plik z raportem w katalogu wskazanym na konsoli.

### Szczegółowy wynik działania programu

Narzędzie Lynis podczas pracy generuje w czasie rzeczywistym szczegółowe informacje odnośnie stanu systemu. Wyświetla użytkownikowi sugerowane działania, które warto podjąć wraz z URL w celu uzyskania dalszych informacji o danej sugestii.



The screenshot shows a terminal window titled "Suggestions (48)". It lists 48 items, each starting with a star symbol and a brief description followed by a URL. The items cover various security-related topics such as updates, PAM configuration, GRUB boot loader, system services, password hashing, and account expiration. A stylized white dragon logo is visible on the right side of the window.

```
* This release is more than 4 months old. Check the website or GitHub to see if there is an update available. [LYNIS]
  https://ciscofy.com/lynis/controls/LYNIS/
* Install libpam-tmpdir to set $TMP and $TMPDIR for PAM sessions [DEB-0280]
  https://ciscofy.com/lynis/controls/DEB-0280/
* Install apt-listbugs to display a list of critical bugs prior to each APT installation. [DEB-0810]
  https://ciscofy.com/lynis/controls/DEB-0810/
* Install apt-listchanges to display any significant changes prior to any upgrade via APT. [DEB-0811]
  https://ciscofy.com/lynis/controls/DEB-0811/
* Install needrestart, alternatively to debian-goodies, so that you can run needrestart after upgrades to determine which daemons are using old versions of libraries and need restarting. [DEB-0831]
  https://ciscofy.com/lynis/controls/DEB-0831/
* Install fail2ban to automatically ban hosts that commit multiple authentication errors. [DEB-0880]
  https://ciscofy.com/lynis/controls/DEB-0880/
* Set a password on GRUB boot loader to prevent altering boot configuration (e.g. boot in single user mode without password) [BOOT-5122]
  https://ciscofy.com/lynis/controls/BOOT-5122/
* Consider hardening system services [BOOT-5264]
  - Details : Run '/usr/bin/systemd-analyze security SERVICE' for each service
  https://ciscofy.com/lynis/controls/BOOT-5264/
* If not required, consider explicit disabling of core dump in /etc/security/limits.conf file [KRNL-5820]
  https://ciscofy.com/lynis/controls/KRNL-5820/
* Configure password hashing rounds in /etc/login.defs [AUTH-9230]
  https://ciscofy.com/lynis/controls/AUTH-9230/
* Install a PAM module for password strength testing like pam_cracklib or pam_passwdqc [AUTH-9262]
  https://ciscofy.com/lynis/controls/AUTH-9262/
* When possible set expire dates for all password protected accounts [AUTH-9282]
  https://ciscofy.com/lynis/controls/AUTH-9282/
* Configure minimum password age in /etc/login.defs [AUTH-9286]
  https://ciscofy.com/lynis/controls/AUTH-9286/
```

### Przykładowe obszary systemu wyświetcone w raporcie:

- wersja systemu ( informacje ogólne)
- rozruch
- jądro systemu
- pamięć, grupy i uwierzytelnianie
- powłoki
- system plików
- urządzenia, sterowniki
- NFS
- usługi
- porty i pakiety
- informacje o sieci
- drukarki
- aplikacje do poczty elektronicznej i wiadomości

- zapora sieciowa ( firewall)
- serwer web
- wsparcie dla SSH
- wsparcie dla SNMP
- bazy danych
- usługi LDAP
- PHP
- logging
- wirtualizacja
- kontenery
- uprawnienia plików

Poniżej znajdują się przykładowe obszary przetestowane przez narzędzie wraz z umieszczonym stanem ich zabezpieczenia:

### Rozruch

```
[+] Boot and services
- Service Manager
- Checking UEFI boot
- Checking presence GRUB2
- Checking for password protection
- Check running services (systemctl)
  Result: found 17 running services
- Check enabled services at boot (systemctl)
  Result: found 17 enabled services
- Check startup files (permissions)
- Running 'systemd-analyze security'
  - ModemManager.service: [ MEDIUM ]
  - NetworkManager.service: [ EXPOSED ]
  - colord.service: [ EXPOSED ]
  - cron.service: [ UNSAFE ]
  - dbus.service: [ UNSAFE ]
  - emergency.service: [ UNSAFE ]
  - getty@tty1.service: [ UNSAFE ]
  - haveged.service: [ PROTECTED ]
  - lightdm.service: [ UNSAFE ]
  - lymis.service: [ UNSAFE ]
  - ntpsec-rotate-stats.service: [ UNSAFE ]
  - ntpsec-systemd-netif.service: [ UNSAFE ]
  - ntpsec.service: [ UNSAFE ]
  - plymouth-start.service: [ UNSAFE ]
  - polkit.service: [ UNSAFE ]
  - rc-local.service: [ UNSAFE ]
  - rescue.service: [ UNSAFE ]
  - rpc-gsssd.service: [ UNSAFE ]
  - rpc-statd-notify.service: [ UNSAFE ]
  - rpc-svcgsssd.service: [ UNSAFE ]
  - rtkit-daemon.service: [ MEDIUM ]
  - smartmontools.service: [ UNSAFE ]
  - ssh.service: [ UNSAFE ]
  - systemd-ask-password-console.service: [ UNSAFE ]
  - systemd-ask-password-plymouth.service: [ UNSAFE ]
  - systemd-ask-password-wall.service: [ UNSAFE ]
  - systemd-fsckd.service: [ UNSAFE ]
  - systemd-initctl.service: [ UNSAFE ]
  - systemd-journald.service: [ PROTECTED ]
  - systemd-logind.service: [ PROTECTED ]
  - systemd-networkd.service: [ PROTECTED ]
  - systemd-rfkill.service: [ UNSAFE ]
  - systemd-udevd.service: [ MEDIUM ]
  - udisks2.service: [ UNSAFE ]
  - upower.service: [ PROTECTED ]
  - user@1000.service: [ UNSAFE ]
  - virtualbox-guest-utils.service: [ UNSAFE ]
```

## Uprawnienia do plików

```
[+] File Permissions
- Starting file permissions check
  File: /boot/grub/grub.cfg
  File: /etc/crontab
  File: /etc/group
  File: /etc/group-
  File: /etc/hosts.allow
  File: /etc/hosts.deny
  File: /etc/issue
  File: /etc/issue.net
  File: /etc/motd
  File: /etc/passwd
  File: /etc/passwd-
  File: /etc/ssh/sshd_config
  Directory: /root/.ssh
  Directory: /etc/cron.d
  Directory: /etc/cron.daily
  Directory: /etc/cron.hourly
  Directory: /etc/cron.weekly
  Directory: /etc/cron.monthly

[ SUGGESTION ]
[ SUGGESTION ]
[ OK ]
[ SUGGESTION ]
[ OK ]
[ SUGGESTION ]
```

## Informacje o sieci

```
[+] Networking
- Checking IPv6 configuration
  Configuration method
  IPv6 only
- Checking configured nameservers
  - Testing nameservers
    Nameserver: 192.168.1.1
    - Minimal of 2 responsive nameservers
  - Checking default gateway
  - Getting listening ports (TCP/UDP)
  - Checking promiscuous interfaces
  - Checking waiting connections
  - Checking status DHCP client
  - Checking for ARP monitoring software
  - Uncommon network protocols

[ ENABLED ]
[ AUTO ]
[ NO ]

[ OK ]
[ WARNING ]
[ DONE ]
[ SKIPPED ]
[ OK ]
[ OK ]

[ NOT FOUND ]
[ 0 ]
```

## Dodatkowe tryby i testy

Narzędzie to umożliwia przeprowadzenie szybkiego skanowania, ograniczenia działania programu do danego katalogu, wykonywanie tylko wyznaczonych testów oraz testów zdalnych poprzez odpowiednie argumenty. Większość możliwych trybów uruchomienia programu znajduje się w podręczniku.

Przykładowe komendy:

- **lynis audit system** – wykonanie sprawdzania systemu (tryb domyślny)
- **lynis upload-only** – wysłanie raportu twórcom narzędzia

Przykładowe typy skanów:

- **audit system remote <host>** – przeprowadzenie zdalnego skanowania

Przykładowe opcje:

- **--no-colors** – wyłączenie kolorowania
- **--quick (-Q)** – wykonanie szybkiego skanowania (bez czekania na sygnał użytkownika)
- **--quiet (-q)** – uruchom bez pokazywania wyników na ekranie. Uruchamia się też **-Q**
- **--use-cwd** – uruchom od obecnego katalogu
- **--warnings-only** – uruchamia **-q** z wyjątkiem ostrzeżeń

W ramach testu narzędzia Lynis przeprowadziłem dodatkowo:

- skan systemu bez podniesionych uprawnień użytkowania
- pełen skan po aktualizacji systemu

**Skan systemu bez podniesionych uprawnień** został oznaczony na samym początku raportu:

```
[+] Initializing program
#####
#   # NON-PRIVILEGED SCAN MODE
#   #
#####
NOTES:
* Some tests will be skipped (as they require root permissions)
* Some tests might fail silently or give different results
```

Uzyskany wynik różni się od przeprowadzonego testu z pełnymi uprawnieniami. Warte zaznaczenia są notatki które informują, że efekt niektórych testów mogą dać odmienne rezultaty. Można zatem wnioskować, że opcja testów z podniesionymi uprawnieniami oferuje dokładniejszy wynik zdrowia i bezpieczeństwa systemu.

```
Lynis security scan details:
Hardening index : 57 [#####
Tests performed : 240
Plugins enabled : 1

Components:
- Firewall [V]
- Malware scanner [X]

Scan mode:
Normal [ ] Forensics [ ] Integration [ ] Pentest [V] (running non-privileged)

Lynis modules:
- Compliance status [?]
- Security audit [V]
- Vulnerability scan [V]

Files:
- Test and debug information : /home/kali/lynis.log
- Report data : /home/kali/lynis-report.dat
```

Umieszczona jest też informacja o pominiętych testach:

```
Skipped tests due to non-privileged mode
BOOT-5108 - Check Syslinux as bootloader
BOOT-5109 - Check rEFInd as bootloader
BOOT-5116 - Check if system is booted in UEFI mode
BOOT-5140 - Check for ELILO boot loader presence
AUTH-9216 - Check group and shadow group files
AUTH-9229 - Check password hashing methods
AUTH-9252 - Check ownership and permissions for sudo configuration files
AUTH-9288 - Checking for expired passwords
FILE-6368 - Checking ACL support on root file system
PKGS-7390 - Check Ubuntu database consistency
PKGS-7392 - Check for Debian/Ubuntu security updates
FIRE-4508 - Check used policies of iptables chains
FIRE-4512 - Check iptables for empty ruleset
FIRE-4513 - Check iptables for unused rules
FIRE-4540 - Check for empty nftables configuration
FIRE-4586 - Check firewall logging
CRYP-7930 - Determine if system uses LUKS block device encryption
CRYP-7931 - Determine if system uses encrypted swap
```

Test po aktualizacji systemu pokazał wynik **59/100**:

```
Lynis security scan details:

Hardening index : 59 [ ##### ]
Tests performed : 258
Plugins enabled : 1

Components:
- Firewall      [V]
- Malware scanner [X]

Scan mode:
Normal [V] Forensics [ ] Integration [ ] Pentest [ ]

Lynis modules:
- Compliance status [?]
- Security audit [V]
- Vulnerability scan [V]

Files:
- Test and debug information : /var/log/lynis.log
- Report data : /var/log/lynis-report.dat
```

Po wynikach można wnioskować, że sama aktualizacja systemu nie gwarantuje poprawy bezpieczeństwa systemu.

Korzystanie z narzędzia Lynis bez instalacji

Narzędzie można stosować z plików pobranych z platformy GitHub. W tym celu potrzebny będzie program **git**.

**Uwaga:** aby skorzystać z tej opcji należy usunąć wersję zainstalowaną przez managera APT

Usuwanie pakietu:

```
sudo apt remove lynis  
sudo apt autoremove
```

```
└─$ sudo apt remove lynis  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following package was automatically installed and is no longer required:  
  menu  
Use 'sudo apt autoremove' to remove it.  
The following packages will be REMOVED:  
  lynis  
0 upgraded, 0 newly installed, 1 to remove and 1 not upgraded.  
After this operation, 1,698 kB disk space will be freed.  
Do you want to continue? [Y/n] y  
(Reading database ... 391171 files and directories currently installed.)  
Removing lynis (3.0.8-1.1) ...  
Processing triggers for desktop-file-utils (0.26-1) ...  
Processing triggers for man-db (2.11.2-1) ...  
Processing triggers for mailcap (3.70+nmu1) ...  
Processing triggers for kali-menu (2022.4.1) ...  
  
└─(kali㉿kali)-[~]  
└─$ sudo apt autoremove  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following packages will be REMOVED:  
  menu  
0 upgraded, 0 newly installed, 1 to remove and 1 not upgraded.  
After this operation, 1,529 kB disk space will be freed.  
Do you want to continue? [Y/n] y  
(Reading database ... 391059 files and directories currently installed.)  
Removing menu (2.1.49) ...  
Processing triggers for doc-base (0.11.1) ...  
Processing 1 removed doc-base file...  
Processing triggers for man-db (2.11.2-1) ...  
Processing triggers for kali-menu (2022.4.1) ...
```

Uzyskanie narzędzia poprzez program git:

```
sudo git clone https://github.com/CISOfy/lynis
```

```
└─(kali㉿kali)-[~]  
└─$ sudo git clone https://github.com/CISOfy/lynis  
Cloning into 'lynis' ...  
remote: Enumerating objects: 14638, done.  
remote: Counting objects: 100% (44/44), done.  
remote: Compressing objects: 100% (31/31), done.  
remote: Total 14638 (delta 21), reused 30 (delta 13), pack-reused 14594  
Receiving objects: 100% (14638/14638), 7.77 MiB | 16.96 MiB/s, done.  
Resolving deltas: 100% (10787/10787), done.
```

W katalogu **lynis** znajduje się plik wykonywalny o tej samej nazwie, który uruchamia narzędzie.

Wykorzystanie narzędzia umieszczonego w katalogu **/lynis/**:

```
(kali㉿kali)-[~]
$ ls
Desktop Documents Downloads lynis lynis.log lynis-report.dat Music Pictures Public Templates Videos

(kali㉿kali)-[~]
$ cd lynis

(kali㉿kali)-[~/lynis]
$ ls
CHANGELOG.md CONTRIBUTORS.md developer.prf HAPPY_USERS.md LICENSE plugins SECURITY.md
CODE_OF_CONDUCT.md db extras include lynis README TODO.md
CONTRIBUTING.md default.prf FAQ INSTALL lynis.8 README.md

(kali㉿kali)-[~/lynis]
$ ./lynis -q

[WARNING]: Test DEB-0001 had a long execution: 32.184039 seconds
find: '/usr/lib/mysql/plugin/auth_pam_tool_dir': Permission denied
[WARNING]: Test PKGS-7345 had a long execution: 17.930268 seconds
[WARNING]: Test CRYP-7902 had a long execution: 26.508159 seconds
Cannot initialize device-mapper, running as non-root user.
Cannot initialize device-mapper, running as non-root user.
pgrep: pattern that searches for process name longer than 15 characters will result in zero matches
Try 'pgrep -f' option to match against the complete command line.

(kali㉿kali)-[~/lynis]
$ ./lynis --version
3.0.8

(kali㉿kali)-[~/lynis]
$
```

Pobrała wersja programu poprzez program **git** to 3.0.8

## Podsumowanie

Lynis może być używany przez różnych użytkowników, takich jak administratorzy systemów, deweloperzy, audytorzy IT i testerzy penetracji do audytów bezpieczeństwa, testów penetracji, wykrywania słabych punktów i sprawdzania zgodności.

Narzędzie wykonuje większość zadań automatycznie a użytkowanie go nie wymaga specjalistycznej wiedzy. Ilość testów, które oferuje oraz licencjonowanie GNU świadczą o jakości tego narzędzia.

Lynis posiada również płatną wersję Enterprise, która oferuje dodatkowe wtyczki rozszerzające funkcjonalność programu (np. o obsługę Docker, Crypto).