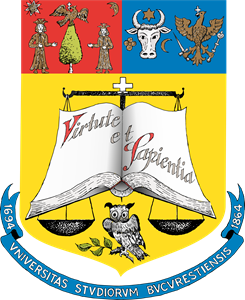
True Nature

Analizator in timp real al securitatii website-urilor accesate



Pirvu Daniel Catalin

Cuprins

[Introducere 3](#_Toc60690494)

[Tipuri de trafic in retele de calculatoare 4](#_Toc60690495)

[Domain Name System (DNS) 4](#_Toc60690496)

[Hypertext Transfer Protocol (HTTP) 6](#_Toc60690497)

[Utilitare in analiza traficului in retele de calculatoare 8](#_Toc60690498)

[Ce este un sniffer? 8](#_Toc60690499)

[Wireshark / Tshark / Pyshark 9](#_Toc60690500)

[Utilitare in analiza securitatii websiteurilor 10](#_Toc60690501)

[Wapiti 10](#_Toc60690502)

[Skipfish / Arachni 13](#_Toc60690503)

[True Nature 14](#_Toc60690504)

[Descriere 14](#_Toc60690505)

[Cum functioneaza? 14](#_Toc60690506)

[Cum se instaleaza? 16](#_Toc60690507)

[Cum se foloseste? 16](#_Toc60690508)

[Concluzii 17](#_Toc60690509)

[Tabela de figuri 18](#_Toc60690510)

[Bibliografie 18](#_Toc60690511)

# Introducere

Vedem cum zilnic apar noi si noi tehnologii si unelte de lucru pe care le putem folosi pentru a analiza cat de poate de in detaliu datele pe care le transmitem sau pe care le primim.

De asemenea, apar noi si noi unelete cu care putem verifica securitatea website-urilor pe care le accesam. Acestea sunt utile pentru ambele categorii de personae. In primul rand pentru dezvoltatori pentru ca pot verifica usor ca munca lor este sigura impotriva diferitelor tipuri de atacuri. Pentru hacker acestea fac parte din prima faza de dezvoltare al unui atac, cea a cautarii intense a vulnerabilitatilor care mai apoi sa poata fi exploatate.

Ambele categorii de unelte sunt foarte eficiente si sunt in continua imbunatatire insa le lipseste o proprietate care ar putea eficientiza si spori timpul de lucru. Vorbim de automatizare. Mare parte din aceste tehnologii trebuie agregate intr-o structura mai complexa pentru a ajunge la rezultatul dorit.

True Nature este o astfel de structura ce vine in ajutorul analizatorilor traficului de date si aduce in plus elementul de automatizare si de integrare cu alta tehnologie utila in scanarea vulnerabilitatilor website-urilor.

Ca si in cazul mentionat mai devreme, acesta poate fi folosit in scopuri bune, utilitatea lui fiind analiza automata a securitatii website-urilor pe care le accesam zi de zi. In scop negativ, un hacker ce nu are o anumita tinta clara poate folosi True Nature si astfel prin simpla navigare pe Internet poate afla informatii si posibile tinte.

# Tipuri de trafic in retele de calculatoare

## Domain Name System (DNS)

Unul din cele mai des utilizate protocoale existente in traficul intre retele de calculatoare este Domain Name System sau simplu spus DNS.

Rolul lui este de a translata numele de domeniu intr-o adresa IP fie ea publica sau privata atunci cand vorbim in context local. De ce am avea nevoie de asa ceva? Raspnsul este pentru ca browser-ele pe care le folosim zi de zi folosesc protocolul Internet (IP) si deci nu stiu cum sa lucreze cu un nume de domeniu.

DNS vine in ajutorul utilizatorilor care pot memora mai usor unui anumit website prin URL-ul acestuia, URL care contine numele de domeniu decat o secventa de numere ca 209.165.200.254 sau mai complicata daca vorbim de adrese IPv6.

Pentru a intelege procesul de translatare realizat de DNS trebuie in primul rand sa intelegem componentele implicate:

1. *Recursorul* poate fi perceput atat ca un software cat si ca un server si este responsabil de primirea cererilor de translatare de la celelalte aplicatii precum browser-e. Tot el este responsabil de crearea si trimiterea cererilor aditionale pentru aflarea adresei IP dorite.

2. Serverul d*omeniu radacina* este primul pas din procesul DNS si reprezinta index-ul unui grup mai mare de calculatore, de obicei asemuite prin tipul de activitate sau locatie geografica, ce ne poate filtra si duce mai rapid cererea catre rezultat.

3. *Serverul de nivel inalt* este colectia in care se poate gasi translatarea ultimei parti de domeniu De exemplu in cazul “example.com”, serverul de nivel inalt este “com”.

4. *Serverul autoritar* sau *recursiv.* Serverul autoritar este ultimul hop in aflarea adresei IP si reprezinta colectia de tip dictionar care ne poate intoarce adresa IP atribuita domeniului din cererea noastra initiala. Un server recursiv nu contine adresa IP dorita si astfel poate crea la randul lui cereri DNS catre alte servere.

Cererea de tip DNS numita si DNS Lookup este compusa dintr-o serie de etape ascunse din punctul de vedere al utilizatorului obisnuit.

La accesarea unui domeniu sa zicem “example.com” pachetul ajunge la un recursor.

Acesta creaza cererea si o trimite catre serverul domeniu radacina.

La randul lui raspunde inapoi cu adresa serverului de nivel inalt ce stocheaza domenii din familia celui dorit de utilizator, in cazul exemplului anterior pentru “.com”.

Recursorul trimite acum cererea catre serverul de nivel inalt si primeste adresa IP a serverului autoritar care il poate returna rezultatul final.

Recursorul face cererea catre serverul autoritar care intoarce adresa IP asociata domeniului de la care am plecat.

Ultimi pasi ar fi ca recursorul ca transmita browser-ului adresa IP pentru a putea continua traficul catre adevarata adresa si stocarea rezultatului intr-un cache local pentru o perioada de timp pentru a eficientiza timpul de raspuns.

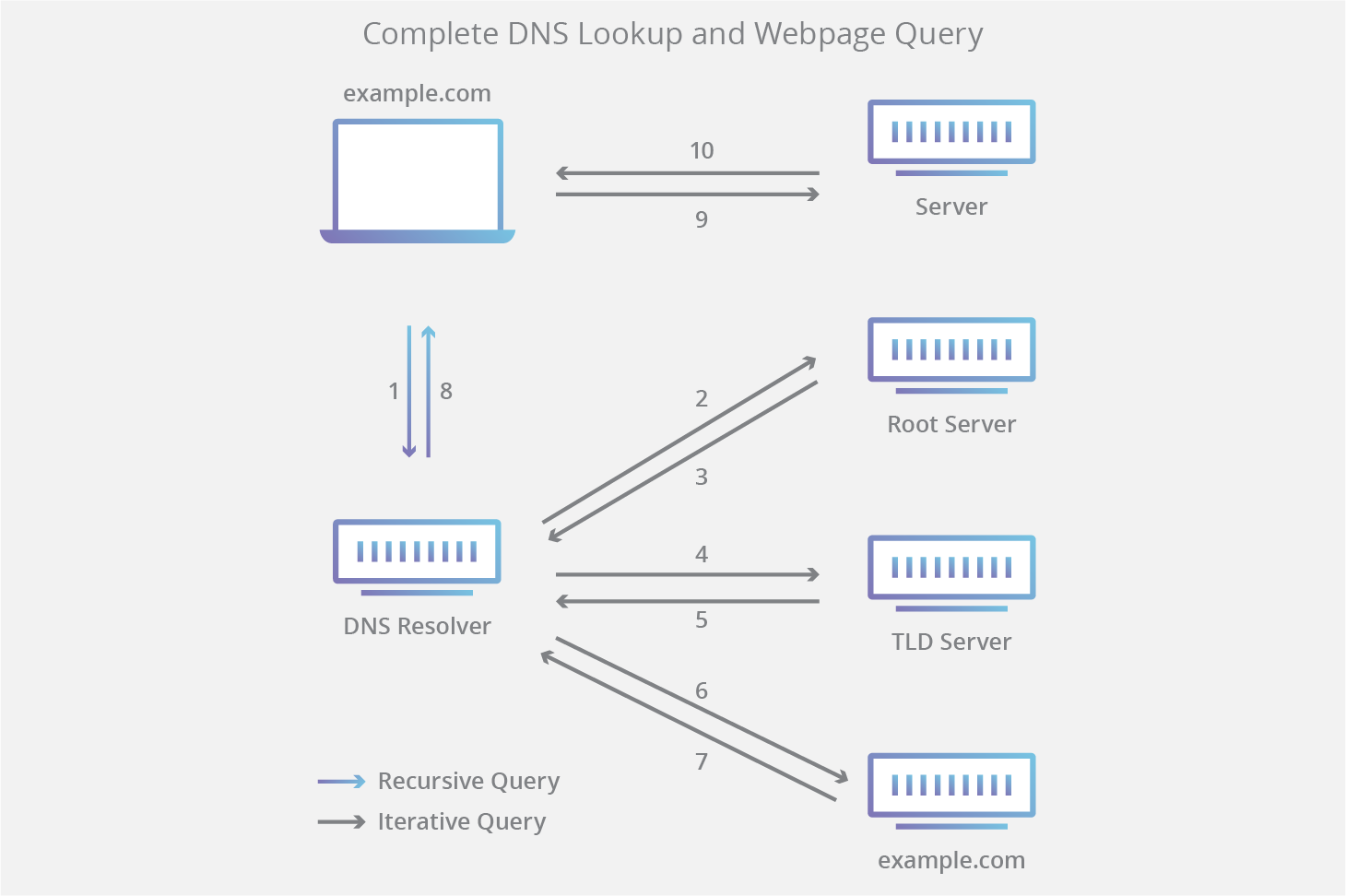


Figure 1 Cerere DNS

## Hypertext Transfer Protocol (HTTP)

Hypertext Transfer Protocol este un protocol de nivel 7 care permite impachetarea si accesul la resurse ce sunt stocate pe servere World Wide Web (WWW). Reprezinta baza a ceea ce inseamna schimb de date in trafic de tip Web.

Este un protocol de tip client-server iar comunicarea dintre cele doua parti se face prin mesaje. Clientul ce este in general un browser, dar poate fi si un utilitar precum curl, trimite cereri catre un server si primeste inapoi mesaje numite raspunsuri. Intre cele doua componenete de regula se afla si altele numite proxy-uri. Acestea au rolul de a opera ca porti de acces sau ca medii de stocare pentru cache.

Clientul sau agentul este cel care intotdeauna initiaza cererile, iar fiecare cerere este trimisa catre un server, dar nu obligatoriu acelasi. Pentru a genera o pagina Web agentul face o cerere pentru a primi de la server structura de baza a paginii in format HTML, dupa ce primeste raspunsul si parseaza fisierul trimite alte cereri pentru elementele aditionale precum stilizare (CSS), functionalitatea (JS) sau obtinere de fisiere media.

HTTP este usor de citit si inteles de catre utilizatori, nu depinde de o stare dar aduce elemente la nivel de sesiune. Pentru ca nu depinde de o stare, nu exista o corelatie intre doua cereri succesive realizate pe acelasi canal. Contextul comun al unei multimi de cereri poate fi insa asigurat de cookies si sesiune.

Transportul este controlat de nivelul inferior ale stivei OSI si iese din aria de actiune al protocolului HTTP. Totusi avem nevoie de transmisie sigura intre cele doua terte participante la trafic (client si server) si astfel ne bazam pe protocolul TCP. El este in prima faza folosit pentru a stabili canalul de comunicare intre cele doua entitati.

Corpul unei cereri contine cateva elemente simplu de inteles: metoda care este in general GET sau POST, locatia resursei, versiunea protocolului, posibile alte date in antetul cererii si , daca vorbim de metoda POST, contine si un corp de date.

Corpul unui raspuns este foarte asemanator cu cel al unei cereri. Elementele cheie sunt codul si mesajul status.

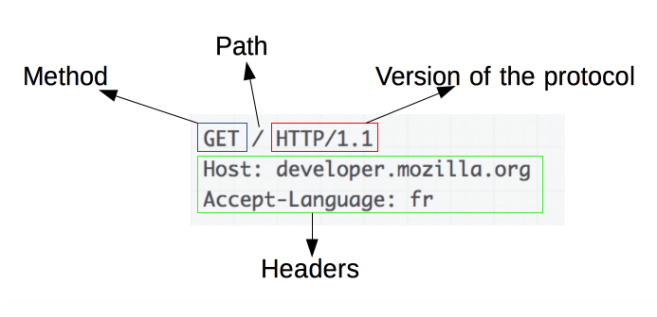


Figure 2 Corp cerere HTTP

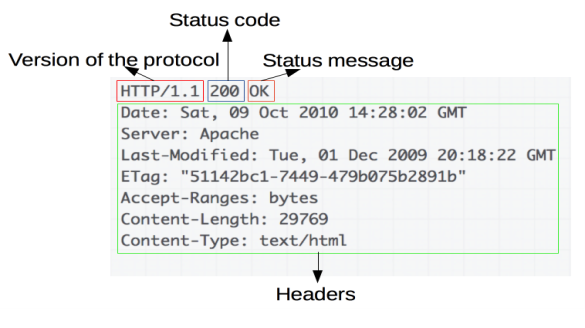


Figure 3 Corp raspuns HTTP

Un procedeu des intalnit in aplicatiile web este redirectarea URL (URL redirection sau URL forwarding). Acesta permite website-urilor sa poate fi accesate prin mai mult de o locatie URL. Procesul este unul simplu, la accesarea unui URL care nu este cel final dorit de dezvoltatori, se va face redirect catre cea finala.

Procedeul se poate aplica si colectiilor ample, vorbim aici de domenii. De exemplu accesarea domeniului ‘facebook.com’ trimite defapt catre accesarea ‘facebook.com’.

De ce am avea nevoie de asa ceva? In primul rand pentru o posibila reducere a lungimii URL-ului si pentru a rezolva mici probleme de scriere (Accesarea fcbook.com trimite catre scrierea corecta facebook.com). Permite detinerea mai multor domenii sub acelasi proprietar si imbunatateste usor securitatea prin combaterea atacurilor de tip phishing care se bazeaza pe acestea mici diferete insesizabile.

Redirectarea ajuta in primul rand la aflarea URL-ului de baza al website-ului. Astfel putem ajunge pe pagina de start al unui website doar prin introducerea domeniului in URL.

# Utilitare in analiza traficului in retele de calculatoare

## Ce este un sniffer?

Un sniffer este un utilitar ce permite monitorizarea traficului in timp real, capabil de capturarea pachetelor atat trimise cat si primite de la alte dispozitive.

In general ele sunt folosite de administratorii de retea pentru a mentine un flow stabil de date si pentru a vedea atunci cand exista nereguli. De regula primul semn prin care ne putem da seama ca dispozitivul nostru este atacat este traficul intens catre si dinstre dispozitiv si latimea de banda utilizata.

Capabilitatile unui astfel de software nu se extind foarte mult. Un sniffer poate doar captura pachete si poate vedea natura si continutul lor.

Desi utilizat in principal in scopuri bune, un astfel de software poate fi folosit si de catre atacatori. Prin simpla monitorizare acestia pot afla informatii pretioate precum emailuri, mesaje sau informatii mult mai sensibile precum credentiale sau informatii financiare. Locul preferat in care acestia pot actiona este o zona acoperita de o retea WiFi publica si nesecurizata.

Exista doua tipuri de sniffing in functie de natura dispozitivelor pe care le folosim. Daca vorbim despre dispozitive hub care prin natura lor transmit trafic la toate dispozitivele conectate la el, atunci ne putem limita la sniffing de tip pasiv. Daca avansam si folosim switch capabil sa transmita pachetele doar destinatarului caruia a i-a fost trimis pachetul atunci vorbim de sniffing activ. Al doilea tip implica si ca software-ul are nevoie sa injecteze trafic aditional pe retea, lucru ce il face mai usor detectabil.

Pentru a ne proteja de astfel de atacuri trebuie sa ne ferim de retele publice WiFi pentru ca daca noi avem avem acces atunci si un atacator poate la fel de bine. Trebuie sa evitam sa accesam websiteuri si aplicatii nesecurizate. Masura generala de securitate este sa folosim continut criptat, fie ca vorbim de protocolul HTTPS sau mai in detaliu de VPN.

## Wireshark / Tshark / Pyshark

Wireshark este cel mai cunoscut utilitar disponibil in mod gratuit si pentru orice sistem de operare pentru realizarea de astfel de monitorizari ale traficului in retele de calculatoare.

Ce poate sa faca? Este folosit in principal pentru ca afiseaza intr-un format citet informatia binara pe care o capteaza. De asemenea, detine o interfata grafica care aduce o serie vasta de filtre si unelte cu care se pot genera statistici.

Este capabil sa interpreteze o serie de peste doua mii de protocoale mai vechi sau mai noi si apartinand tuturor nivelelor stivei OSI dar in general vor fi de tip TCP, UDP sau ICMP.

Cu toate ca dispune de o monitorizare vasta, el nu trimite alerte daca un anumit tipar este intalnit si de aceea nu trebuie confundat cu un sistem de detectare a intruziunilor (Intrusion Detection System - IDS).

Varianta de terminal a programului Wireshark se numeste Tshark si dispune de aceeasi functionalitate insa trebuie compuse comenzi complexe. Tshark este foarte util atunci cand vrem sa integram monitorizarea pachetelor intr-o structura mai complexa.

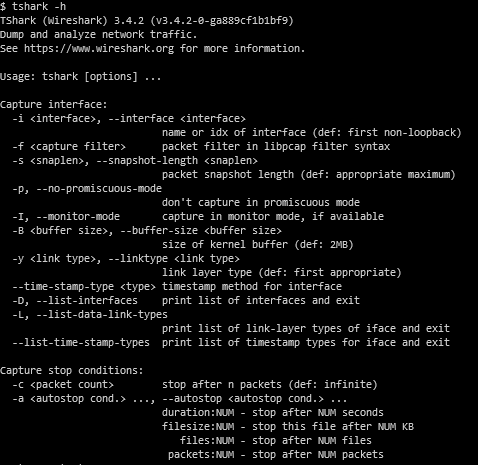


Figure 4 Secventa din rezultatul comenzii "tshark -h"

Daca structura mai complexa alegem sa o construim folosind limbajul de programare Python atunci un utilitar care se foloseste de Tshark este Pyshark.

Toate cele trei utilitare dispun atat de analiza in timp real cat si de incarcarea unui segment de trafic captat anterior.

# Utilitare in analiza securitatii websiteurilor

## Wapiti

Wapiti este un analizator al securitatii websiteurilor sau mai amplu spus al aplicatiilor web. Este gratuit si deoarece are la baza limbajul Python se poate folosi pe mai multe sisteme de operare.

Wapiti lucreaza intr-un blackbox si este independent de codul sursa al aplicatiei web. El primeste un URL de start, dupa care incepe sa parcurga websiteurile aplicatiei in incercarea de a gasi secvente de cod sau formulare unde poate injecta date. Odata ce gaseste toate aceste zone de actiune Wapiti incepe sa injecteze date pentru a testa si afla dar exista vulnerabilitati.

Este capabil sa detecteze vulnerabilitati din multe zone de interes:

* Descoperirea unor fisiere
* Vulnerabilitati ale bazelor de date
* Cross Site Scripting XSS
* Detectia unor metode de evaluare precum eval() sau system()
* Folosirea unor fisiere nesigure (Se bazeaza pentru asta de baza de date al utilitarului Nikto)
* Prezenta unor fisiere de backup ce ar putea arata codul sursa
* Fisiere .htaccess slabe
* Metode HTTP ce pot fi mascate prin PUT
* Etc.

Poate actiona folosing ambele metode principale HTTP (Get si Post). Se poate folosi de formulare complexe si poate chiar injecta date sub forma de fisiere incarcate.

Rezultatul poate fi expus in mai multe forme, cel implicit fiind cel de pagina HTML.

Desi inca din start se poate seta un nivel de afisare al informatiilor cat mai explicit si un timp maxim in care scannerul isi poate face treaba, putem si noi sa oprim sa mai bine spus sa punem pe pauza scanarea pentru a vedea ce a reusit sa obtina pana la momentul respectiv.

Putem apoi alege sa sara peste anumite etape (de exemplu sa nu mai continue sa caute alte pagini si alte posibile zone de actiune sau sa treaca peste un anumit tipar de atacuri) sau chiar sa oprim acolo intregul proces.

Oprirea manuala se poate incheia cu sau fara generarea raportului, pe cand comportamentul implicit atunci cand scanarea isi termina procesul normal este de a genera raport in locatia specificata in comanda sau daca nu se precizeaza in locatia de baza a utilizatorului de sistem ce initiaza scanarea.



Figure 5 Secventa din rezultatul comenzii "wapiti -h"

In urmatoarea poza putem vedea cum Wapiti este pornit si cum afiseaza in terminal fiecare informatie utila utilizatorului pentru ca acesta sa vada parcursul in timp.

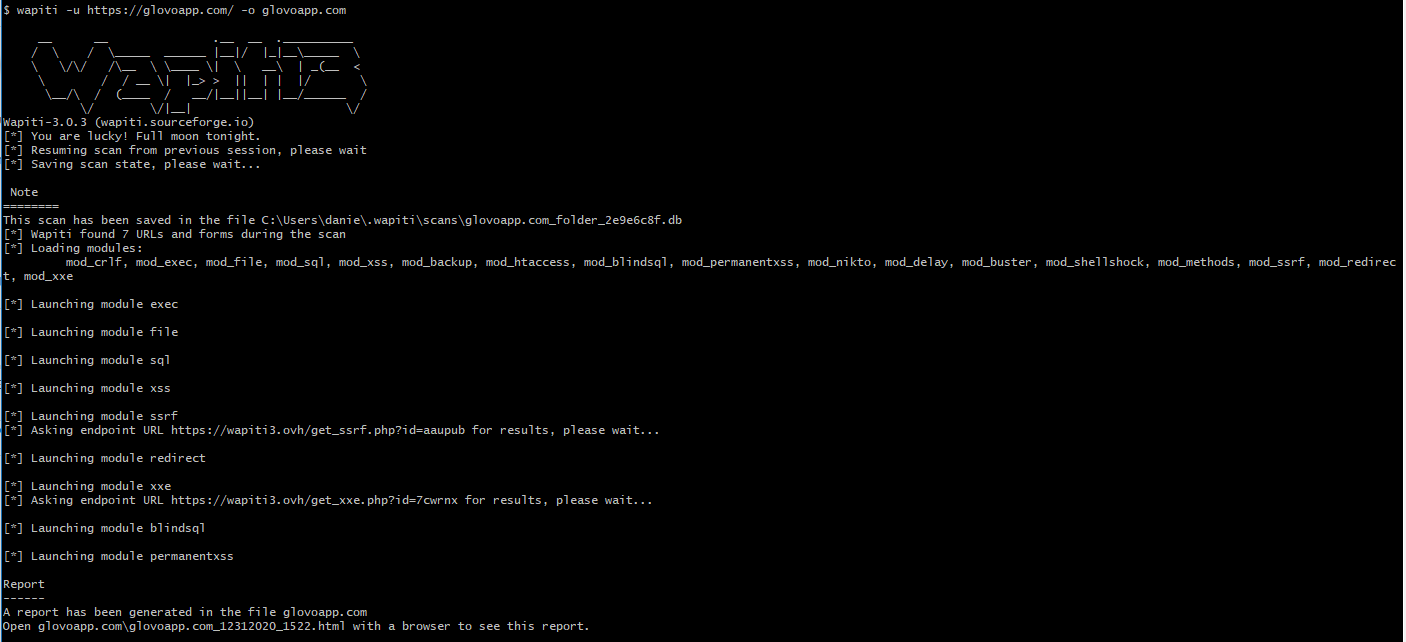


Figure 6 Continutul scanarii Wapiti in terminal

Raportul implicit de tip HTML arata de forma urmatoare.

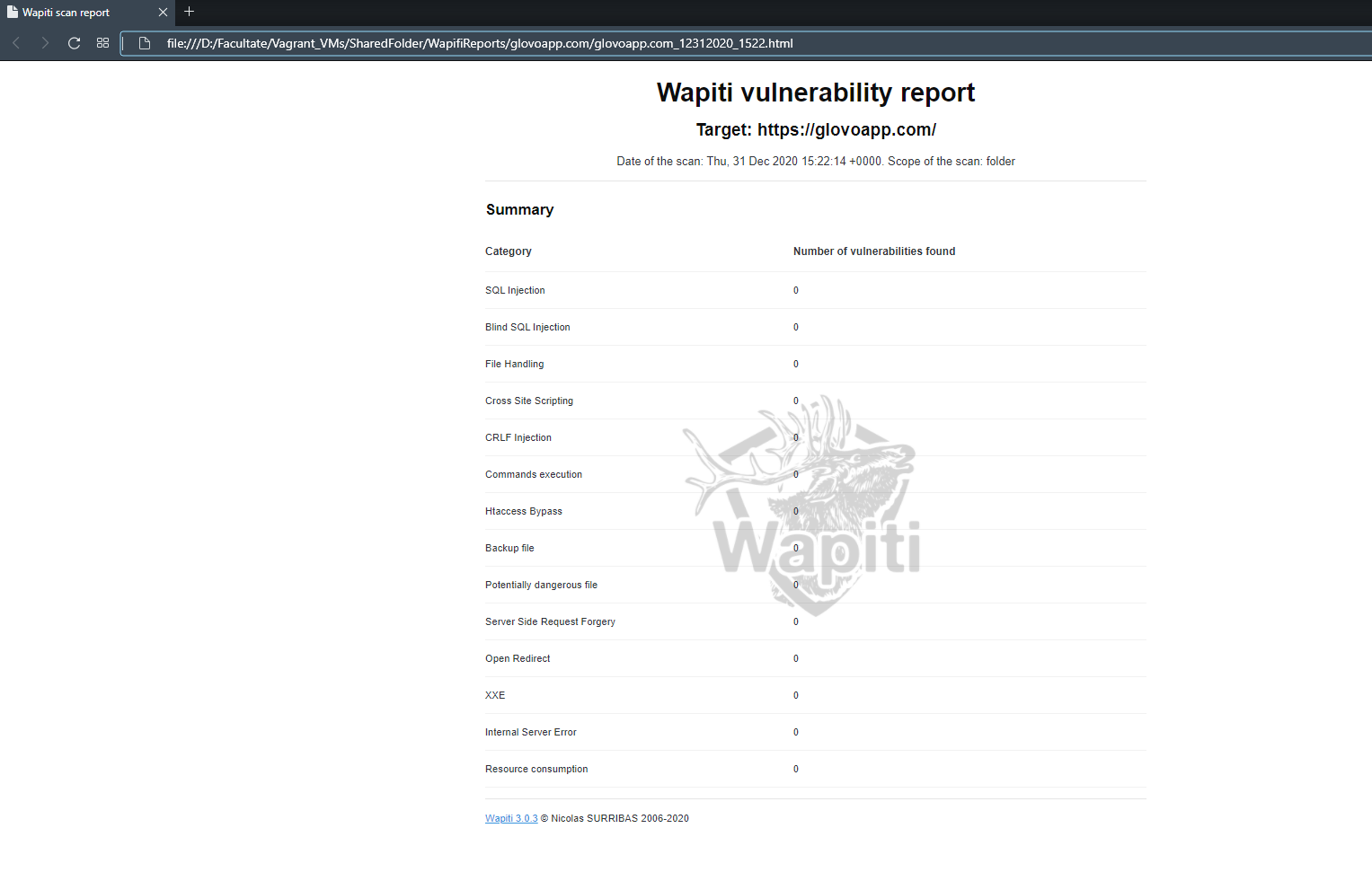


Figure 7 Raport HTML Wapiti

Raportul arata sub forma tabelara numarul de vulnerabilitati gasite (In cazul website-ului testat zero) pentru fiecare categorie de atacuri. Daca ar fi gasit vulnerabilitati raportul ar fi fost mai extins si ar fi aratat si metodele de preventie impotriva atacurilor.

## Skipfish / Arachni

Printre alte utilitare de scanare a securitatii website-urilor se numara Skipfish si Arachi.

Acestea au o desfasurare a actiunilor similara daca nu chiar identica cu Wapiti. Este vorba doar de performanta si de preferinta formatului rezultatului atunci cand alegem ce utilitar sa folosim.

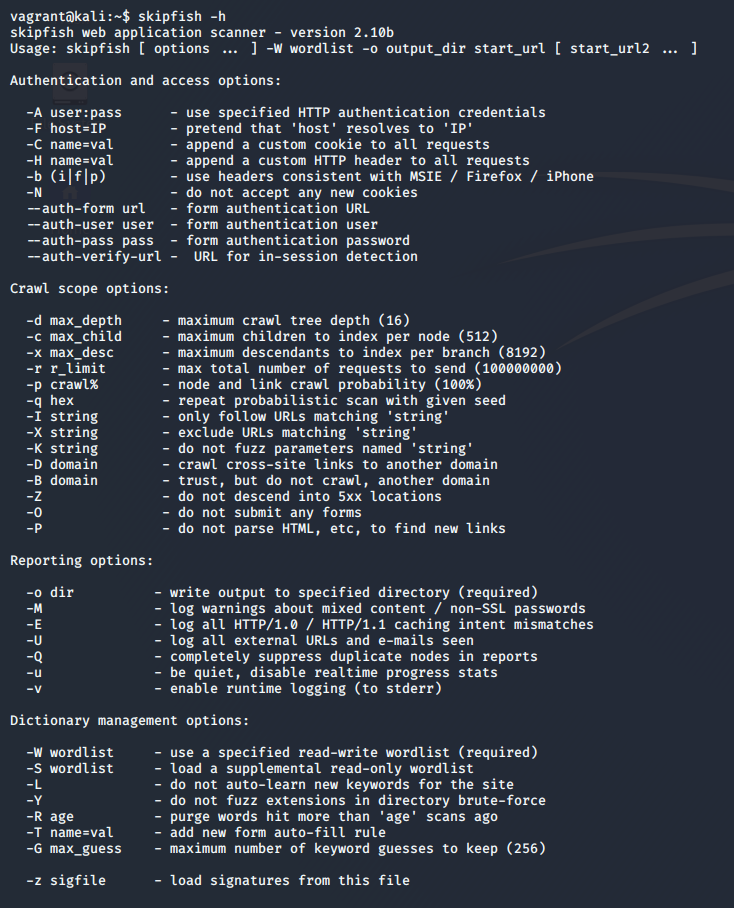


Figure 8 Secventa din rezultatul comenzii "skipfish -h"

# True Nature

## Descriere

True Nature este un utilitar menit sa combine toate tehnologiile descrise pana acum intr-un format simplu de utilizat, automatizat si transparent.

In urma executarii programului, la o locatie aleasa de noi la inceput, vom gasi rapoartele de securitate realizate pentru website-urile pe care navigam.

## Cum functioneaza?

In primul rand programul detecteaza tipul de sistem de operare. Acest lucru este util in verificarea corectitudinii urmatorei decizii ce trebuie realizata de utilizator.

Tot in partea de inceput se preiau informatiile din fisierul de configurare. Aceste configurari includ calea in sistemul de fisiere unde se vor salva rapoartele si doua liste, una care contine secvente de domenii pe care le dorim excluse din traficul captat si una care contine host-uri explicite care trebuie excluse.

Se foloseste Tshark pentru a afla si afisa lista interfetelor de retea pe care le putem folosi. Urmeaza ca utilizatorul prin introducere manuala de la tastatura sa aleaga interfata tinta. In cazul sistemelor Linux trebuie specificat numele explicit al interfetei, pe cand in cazul sistemului Windows trebuie specificat numarul interfetei.

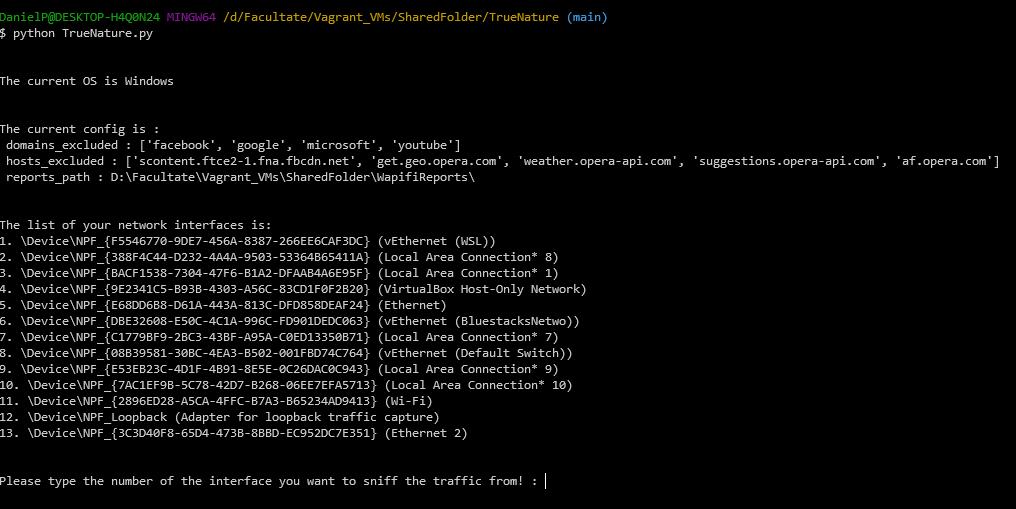


Figure 9 Secventa de alegere a interfetei de lucru

Dupa ce obtinem aceste informatii programul poate incepe executia efectiva.

Se vor realiza captari ale traficului de tip DNS in intervale de timp de cate 5 secunde. Captarea pachetelor se realizeaza cu ajutorul lui Pyshark care primeste atat interfata cat si un filtru cu care putem exclude anumite domenii descoperite, filtru ce este creat automat pe baza celor doua liste din fisierul de configurare.

Dupa fiecare interval, se parcurg domeniile aflate si se incearca accesarea domeniilor folosind protocolul HTTP. Rolul este de a afla prin mecanismul de redirectare adresa URL adevarata pe care o vom folosi la analiza de securitate.

Inainte de pornirea scanarii fiecarui domeniu se verifica sa nu generam rapoarte pentru domenii deja testate.

Daca domeniul este nou, se porneste un nou proces scanarea folosind Wapiti. Deoarece se executa scanari in paralel putem eficientiza timpul general al programului.

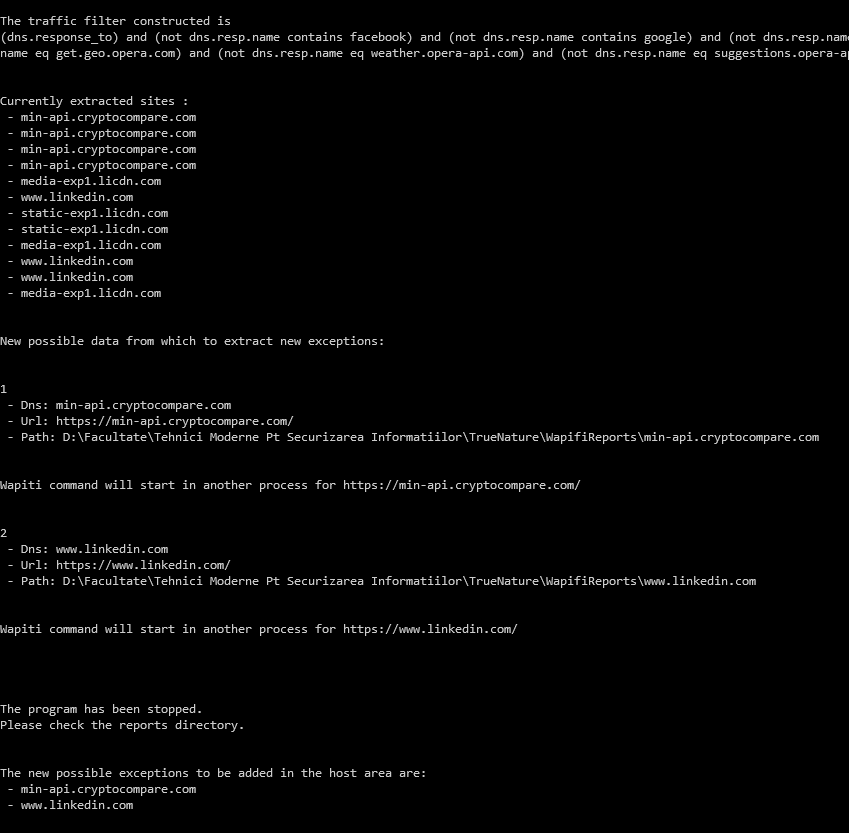


Figure 10 Mesajele afisate in terminal in timpul si dupa terminarea programului

Cu toate acestea, pentru siguranta, programul asteapta ca fiecare domeniu aflat in intervalul de timp sa fie analizat, scurt spus asteapta ca toate procesele paralele sa se termine.

Acest lucru poate creste timpul general de executie pentru ca natura utilitarului Wapiti este de a cauta noi si noi adrese si tinte plecand de la adresa URL pe care o dam ca parametru si astfel se poate ajunge la o structura foarte mare derivata.

Aceste argumente pro si contra sunt date tocmai de utilizarea scanerului de vulnerabilitati in modul implicit( in care aduna posibile zone de testare si din pagini derivate ). Se poate limita Wapiti doar la scanarea website-ului dat ca parametru insa asta ar nu ar genera la fel de multa incredere in scanarea securitatii aplicatiei web per total.

## Cum se instaleaza?

Programul este scris folosind limbajul de programare Python si de aceea este necesara instalarea lui pe sistemul de operare. Recomandarea este de a instala ambele versiuni (atat Python 2 cat si Python 3).

De asemenea este necesara instalarea administratorului de module Python numit *pip.* Pentru sistemul Windows vine preinstalat cu Python insa pentru sisteme baza pe Linux trebuie instalat separat.

Inainte de a trece la instalarea modulelor avem nevoie de Wireshark. Acesta se poate instala in cazul Windows ca un program executabil

Avand utilitarul pip instalat, putem instala modulele necesare care sunt Pyshark, PyYAML.

Instalarea analizatorului Wapiti se realizeaza diferit. Pentru Windows, trebuie descarcata arhiva de pe pagina oficiala si apoi utilitarul se poate instala folosind un terminal cu drepturi de autor cu ajutorul comenzii *python setup.py install.* Pentru sisteme Linux se realizeaza mai simplu cu ajutoul administratorului de pachete de pe respectiva distributie. De asemenea pe Kali Linux este preinstalat.

## Cum se foloseste?

Dupa ce avem toate cele necesare instalate putem folosi True Nature ca un program obisnuit scris in Python. Pentru a-l porni folosim comanda *python TrueNature.py* sau *python3 TrueNature.py* dupa caz.

Daca nu avem setata o locatie unde sa se salveze rapoartele atunci vom primi un mesaj iar programul se opreste. In fisierul de configurare *Config.yml* trebuie setata calea absoluta catre un director existent. Trebuie sa avem grija ca la finalul sirului de caractere sa marcam ca este vorba despre un director prin adaugarea unui simbol slash (exemplu D:\TrueNature\WapifiReports\).

Daca avem configuratia realizata, programul ne va intreba de pe ce interfata sa captureze trafic.

Dupa ce alegem programul incepe sa lucreze fara a mai fi nevoie de interactiunea noastra.

Programul se opreste doar manual prin secventa de taste Ctrl + C . Dupa oprire se afiseaza lista de posibile inregistrari pe care manual le putem adauga in fisierul de configurare pentru a reduce numarul de pachete capturate si pentru a face lucrurile mai usoare pentru celelalte utilitare implicate.

# Concluzii

Plecand de la faptul ca True Nature isi doreste a fi un program ce realizeaza analiza de securitate si foloseste utilitare doar in acest sens nu trebuie confundat un un sistem de detectie. El reprezinta o automitazare si o simplificare a unui proces recurent si realizat in mod normal manual implicand multe utilitare, fiecare cu scop diferit.

Natura elementelor folosite il face, de asemenea, sa poata fi utilizat si in scopuri negative atunci cand un atacator nu doreste sa se grabeasca in alegerea unei anumite tinte. Aici insa procesul consider ca ar putea fi realizat mult mai eficient fara implicarea programului True Nature.

De mentionat este si faptul ca toate tehnologiile si utilitarele folosite sunt in continua imbunatatire. Astfel instalarea noilor versiuni ale elementelor folosite implica automat imbunatatirea programului True Nature. De asemenea, pentru ca programul este modularizat putem alege sa schimbam anumite structuri, de exemplu sa folosim alt analizator de securitate web sau sa realizam capturarea pachetelor bazandu-ne pe un context mai general.

# Tabela de figuri

[Figure 1 Cerere DNS 5](#_Toc60693347)

[Figure 2 Corp cerere HTTP 7](#_Toc60693348)

[Figure 3 Corp raspuns HTTP 7](#_Toc60693349)

[Figure 4 Secventa din rezultatul comenzii "tshark -h" 9](#_Toc60693350)

[Figure 5 Secventa din rezultatul comenzii "wapiti -h" 11](#_Toc60693351)

[Figure 6 Continutul scanarii Wapiti in terminal 12](#_Toc60693352)

[Figure 7 Raport HTML Wapiti 12](#_Toc60693353)

[Figure 8 Secventa din rezultatul comenzii "skipfish -h" 13](#_Toc60693354)

[Figure 9 Secventa de alegere a interfetei de lucru 14](#_Toc60693355)

[Figure 10 Mesajele afisate in terminal in timpul si dupa terminarea programului 15](#_Toc60693356)

# Surse figuri

Figure 1 - <https://www.cloudflare.com/learning/dns/what-is-dns/>

Figure 2 - <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>

Figure 3 - <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>

# Bibliografie

1. <https://www.cloudflare.com/learning/dns/what-is-dns/>
2. <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview>
3. <https://en.wikipedia.org/wiki/URL_redirection>
4. <https://www.avast.com/c-sniffer>
5. <https://www.avg.com/en/signal/what-is-sniffer>
6. <https://www.csoonline.com/article/3305805/what-is-wireshark-what-this-essential-troubleshooting-tool-does-and-how-to-use-it.html>
7. <https://wapiti.sourceforge.io>
8. <https://tools.kali.org/web-applications/skipfish>
9. <https://www.wireshark.org/docs/>
10. <https://www.wireshark.org/docs/man-pages/tshark.html>
11. <https://kiminewt.github.io/pyshark/>