

Information Gathering: frame the target with IG

Claudio Rimensi
University of Padua,
Computer and Network Security

February 18, 2019

Abstract

Research of information represents an important aspect for every penetration tester. Thanks to the collection of information, it's possible to get data or attack a specific target. This phase can be considered to be the first part for every type of cyber-attack. Without this step, nothing can be done. In this paper I want to explain a solution, called IG (information Gathering), where I tried to obtain information about web services. It is part of a larger program, called Perseus, whose purpose tries to embrace a good portion of the attacks that can negatively affect technologies with minimum participation of the human. However, Perseus will not be treated because it is still under development. The program will be show in action and compared with other similar programs.

1 Introduction

What is Information gathering? Information gathering helps the individual and the organization to undertake complicated tasks that would otherwise be extremely hard to accomplish and unfeasible without the benefit of gathered information. As defined in the dictionary, information gathering is the act of collecting information from various sources through various means. Information gathering is the responsibility of the research specialist within the organization's intelligence department. They are the personnel properly trained and equipped to carry out the research tasks in the most efficient manner. The proper handling of data requires unique methods and procedures in the field of information gathering. Research personnel do this task unequivocally through skills like data sifting, intelligent questioning and other research skills. Other company personnel can also do their own information gathering on the personal level to improve their job performances and as a self-help tool. Researchers undertake information gathering in order to:

- Broaden the scope of knowledge of the organization
- For the development of particular skills
- To reduce the apprehension caused by the unknown
- For a higher level of understanding of special subjects
- Solving problems

Information Gathering can be divided up into 3 parts:

1. Footprinting
2. Scanning
3. Enumeration

The first step, called footprinting, represents the phase in which the target is framed by passive research, acquiring all that is needed. At the end of the process, if everything has been done correctly, you have a footprint, that is a unique profile of the target. The second step is called scanning. In this phase you can scan with particular tools, the port or OS of the web server. Then, the last step is enumeration. Thanks to this passage an attacker can examine in depth the identified services looking for further weakness. Finally, these three procedures are similar, but footprinting is a method to pick up information using passive techniques. While scanning and enumeration uses active techniques. So it is necessary implemented tool, like Proxychains, that make the anonymity the penetration tester during the active attack.

This paper, will describe the information gathering about companies. I will focus on the main Universities of Italy and will present my findings on them.

Therefore I will consider the Italian Universities participating in the event called CyberChallenge[1]. These Universities are: Alma Mater Studiorum (BO), Polytechnic of Milano, Polytechnic of Torino, University of Sannio, University of Bari, Wisdom University of Rome, Università of Cagliari, Università of Salento, University of Calabria, University of Camerino, University of Genova, University of Naples, University of Padua, University of Perugia, University of Pisa, University Politecnica of Marche, Center of competence Cybersecurity Toscano and Link Campus University of Rome.

Specifically, the paper will be outlined as follows: Section 2 related work. Section 3 description and implementation of IG. Section 4 I explain some countermeasures to guarantee a sort of protection by the research attack. Finally in the section 5 I present the tests on a class of targets and respectively results and comparison my solution with others.

2 Related work

With the research in information Retrieval and phenomenal growth of the web, today's websites have become a key communication and information medium for various organizations[2]. It also offers an unprecedented oppor-

tunity and challenges to data mining. Various techniques are available to extract useful data from the web and store them.

In [3] it has tried to improve the accuracy of query result of search engine and satisfy personalized requirements of users con un personalized Search Engineer model. This method based on certain information which mine from users' behaviors and customs in using search engineer.

In [4] has been conducted a survey of how Web content mining plays an efficient tool in extracting structured and semi structured data and mining them into useful knowledge.

In [5] it explains a way to organize data in the web with Text clustering as a method of organizing retrieval results can organize large amounts of web search into a small number of clusters in order to facilitate users; quickly browsing. This method implements word clustering by calculating word relativity and then implements text classification.

In [6], [7],[8], [9] and [10] are described in detail the techniques both online and offline, used in the information gathering to get data about people and company. Tutto racchiuso sotto il nome di OSINT (Open Source INTelligence).

OSINT is distinguished from simple search because it applies an information management process in order to create a specific knowledge in support to a specific decision of an individual or group. It is good to underline that "Open Source" in this context has nothing to do with the meaning of the same term referring to "open source" software.

Finally in the [11] a short article is presented where we talk about the defense in depth to ensure that the companies can deal and manage the potential risks that come from the network and threaten to compromise their security and business, must effectively understand all the real risks they can face, whose size is based on the value of their assets.

3 Description and implementation of IG



Figure 1: Initial phase where the user put the domain

Before starting to execute IG, I decided to explain how IG works. Below there is a scheme, figure 2, that describe the operations done by IG. It is important to remember that IG invokes other scripts or uses particular URL to get the specific information. IG has been developed in the Linux environment using Ubuntu 18.04. Code for the system has been written in Python, in Bash and in C.

The figure 1 shows the first step, where the user put the name of domain.

From the figure 2, IG presents 19 steps to get the info from web services.

I decided to organize it in this way. For each step, I write what it is and how did I implement in a real environment. The bash file recalled by IG have arguments domain and name of file, where insert the information of that specific operation.

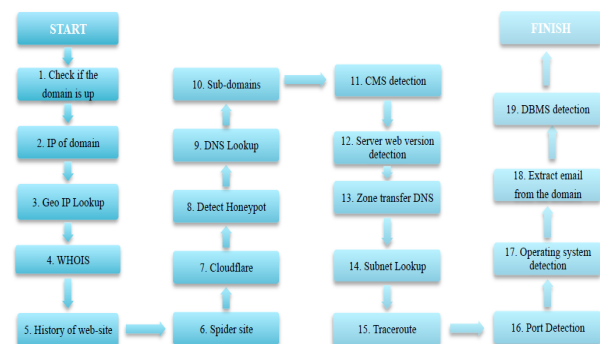


Figure 2: Scheme of IG

1. Check if the domain is up

Description

The study of target will begin from examining whether the web service is active or not. If it is enable the message will be "HTTP/1.1 200 OK", where "/1.1" is the protocol's version.

Implementation

I used a function belonging to the socket library of Python. "socket.gethostbyname(dominio)". If doesn't exist domain written, I inserted an error socket.gaierror that tell me the domain is not available and it will open a web-site <https://hostingchecker.com> where the user can verify if the URL is protect by Reverse Proxy.

2. IP domain

Description

IG finds the IP of domain, useful for subsequent attacks.

Implementation I used a function belonging to the socket library of Python. "socket.gethostbyname(dominio)". Thanks to function I get the IP address.

3. Geo IP lookup

Description and implementation

IG tries to find the latitude and longitude of web server, in such a way to find the position of web service. For this point I built two solutions. The first I exploited this site <http://api.hackertarget.com/geoIP>.

The second solution is more simple, with regards to the implementation, but it is not automatic. After, the first solution, IG run to the second solution, where IG open 3 pages of internet. The first is the main page of Google, where in the search bar there is IP of service web, in the second page there is this specific site <https://www.maxmind.com/en/geoip2-precision-demo>, when IG finished its work, the user puts the IP in a box and presses ENTER compare coordinate in a table. Lastly the user takes coordinates and puts them in the third page, where there is Google Maps open and so it find the position of web server.

4. Whois

Description

Among the basic profiling tests that IG does, there is WHOIS LOOKUP to get public info. **Implementation**

I used the command "WHOIS", followed by the domain.

5. History

Description and implementation

After IG opens a page on browser, which compares the history of domain. This page is: <https://web.archive.org>. I included this part in a bash that was recalled from IG. It is very useful, because it helps the attacker to know others information of the site. In particular, if someone has taken away data in the current version, like email address, telephone number ecc, in the previous version, maybe these information could be important for future attacks.

6. Spider site

Description

Afterwards IG starts a crawler, in the specifically a crawler web.

Implementation

For this scope, I built this solution in a bash file recalled by IG. In this file to obtain the list of URL I use a tool called wget.

7. CloudFlare

Description

Sometimes the IP address may not be the real address of the server, as there are many who rely on Reverse proxy. The web hosting companies, to save the hardware costs, run multiple web server within the same machine: with intermediate infrastructure, this means all those system facing the resolution of the real IP address of the machine. An example could be CloudFlare.

Implementation

For Cloudflare I used a code, downloaded from Github repository called theLinuxChoice, described in the related work section2. I took just the part of Cloudflare and I put the code in a bash file.

8. HoneyPot

Description and implementation

In this pass IG can find if it presents a honeypot or not. Concerning honeypot I exploit this URL <https://api.shodan.io/> to determine the probability of detect a real system or not. The defining characteristics of known honeypots were extracted and used to create a tool to let you identify honeypots! The probability that an IP is a honeypot is captured in a "Honeyscore" value that can range from 0.0 to 1.0.

9. DNS Lookup

Description

IG tries to get the list of DNS record of the specific domain. The request is made through DNS server.

Implementation

I implemented the detection of server web version using a command of Linux called HOST, followed by the domain.

10. sub-domains

Description

IG search subdomain of the target and put every result into the file that I created.

Implementation

IG exploits exploited a script downloaded from this web-site <https://github.com/christophetd/censys-subdomain-find>.

11. CMS detection

Description

Then IG identifies the CMS. If the web service has been developed through a content management system, then it is possible by way of the recognition of similar patterns to refer to the name and possibly to the version in use. **Implementation**

IG can detect 4 types of CMS: Joomla, Wordpress, Drupal and Opencart. Exists a lot of CMS and so I had in mind to put the CMS that are used more often.

12. Server Web detection

Description

Besides further more determine the web server version through the technique called Banner Grabbing, in which banner is identified the output that the server offers following a simple request to a TCP/IP port listening on the server.

Implementation

I implemented the detection of server web version using a command of Linux called HEAD. Both for server web version detection and traceroute I decided to insert the Proxychains, in order to guarantee a good anonymity.

13. Zone Transfer DNS

Description

DNS Zone transfer is one of the many mechanisms available for administrators to replicate DNS databases across a set of DNS servers. It authorizes a secondary master server to update its zone database from the primary master. This provides redundancy in DNS management, for cases where the primary server is not available. In general, a DNS zone transfer should only be performed by secondary master DNS servers. However, many DNS servers are badly configured and provide a copy of the zone.

Implementation

For implementation I exploited a propriety of a web-site: <http://api.hackertarget.com/zonetransfer/>.

14. Subnet Lookup

Description

IG tries to find the subnet of the target. When working with networks is often required to quickly calculate the network boundaries of a subnet from a CIDR address or subnet with network mask.

Implementation

For implementation I exploited a propriety of a web-site: <http://api.hackertarget.com/subnetcalc>

15. Traceroute

Description

So regarding traceroute, IG uses this tool for displaying the route (path) and measuring transit delays of packets across an Internet Protocol (IP) network.

Implementation

I did traceroute using the command "traceroute". As server web version I used Proxychains to guarantee the anonymity.

16. Port detection

Description

Then IG detect the port of web server.

Implementation

The tool used for this goal is Nmap. Nmap has been configured for IG with very specific functions: The "-PN" option is used to skip the host search phase, saving time, since IG has arrived up to that point. The "-sT" option specifies the connection that is of the TCP type. The TCP connect type scan is the default TCP scan where SYN scanning is not available option. This is the case in which a user does not have privileges on sending "raw" packages. The "-n" option is used to ensure that no DNS requests are made outside the TOR network. The "-sV" option is used to perform the detection of services and versions on each open port, while the "-p" option is used to tell Nmap which ports to scan. To ensure anonymity I used proxychains.

17. OS detection

Description

Part of the success of a cyber-attack is given by

knowing which operating system is installed on the server. Knowing OS allow us to have a clearer idea of the environment that will be attacked. Therefore, the IG starts a procedure for detecting the OS of the web server.

Implementation

The tool used for this goal is Nmap with proxychains.

18. Extract email from domain

Description

Another research's attack of IG is try to extract email from web server, using a particular framework Metasploit.

Implementation

I have implemented Metasploit into a file in bash so it can work autonomously.

19. DBMS detection

Description

Then IG can identify if the server web, which provides the web service, have a port 3306 open, in order to detect the model and version of the DBMS. Here I used, as in the extract email step, the Metasploit framework. **Implementation**

If port 3306 compares open, IG starts the framework, and automatically inserts a series of parameters that are used to identify the model and the version in use of the DBMS.

Below here, there is figure 3 showing the process of analysis of the domain entered by the user.

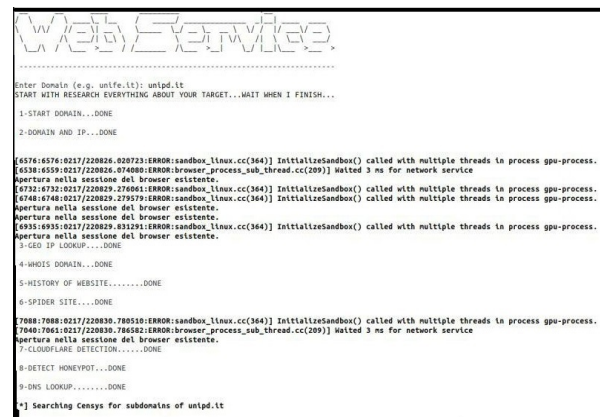


Figure 3: data collection phase

4 Countermeasures

In this section I would like some precautions against the research's attack. Among these, you can find a defense against WHOIS.

In fact, every time a domain is registered, the Registrar must communicate the registration data and make it public (these rules are established by ICANN[12]).

However, the domain owner, to avoid being subjected to spam attacks or serious violations of privacy, may decide to obscure personal data to the public. This function is often sold as an additional package by the Registrar and can only be applied to generic domains (.com, .net) while the territorial domains (.it,.eu,.ed etc) are excluded.

Regarding the techniques for guessing the subdomain that points to the correct IP address, and then using Reverse proxy, the sysadmin should try not to expose the real IP address on the network. Other useful techniques can be related to DNS security and honeypot installation.

As for the network recognition countermeasures, there are several commercial systems for intrusion detection in the network, NIDS (Network Intrusion-Detection Systems) and Intrusion Prevention, IPS (Intrusion Prevention System), which allow to identify Network Recognition Activities. A possibility is to modify the source code of the operating system, or to alter a parameter, but this would lead to system functional problems FreeBSD[13]

5 Evaluation and Comparison

5.1 Test and results

In this section I shows the results get by IG. The Table 1 shows the results about the web services analyzed. In general, I get the same things from the domains, however there were some exceptions.

- Polytechnic di Torino, University of the Salento, University of Camerino, University of Naples present a Reverse Proxy, so IG didn't work.
- Regarding University of Padua, IG has identified a CMS, called Drupal.
- University of Bari Aldo Moro was the university that showed the most info, so much so that IG managed to carry out even a transfer of Zone.
- DBMS, Honeypot were not detected by IG.
- Although CloudFlare was not identified by IG, maybe it is presents in the 4 domains not analyzed by IG.

5.2 Comparison with other solutions

My program isn't unique, in fact exit other solutions but less performance because present a list of operations where the user have to choose what he want to do. There are some positive aspect:

- My solution works in an automatic way, while the others didn't, except[14].

- Saves the results in a file, where each file is saved by the domain's name, while the others didn't, except[14].
- IG's time performance are better than other scripts. Because in particular in the port scanning traceoute and OS system I have put in option to make more quickly the process. The time for scanning for that particular part in my solution is around 1-2 minutes. In the other solutions the average is 4 minutes. In general the total time used from IG is around 7-8 minutes for each domain, while in the others is 12 minutes.
- In [17],[18],[19] and [20], at the end of the process of a operation choosed by user, the program stops and goes out.
- In [15] and [16] are focused only sub domains.
- In [21] and [22] are present operations for basic passive research like IP,coordinate,whois and HTTP header.
- The [14] presents the option "all" where it's possible get info with active and passive attacks and save the results on the file. Is a little slower than IG because it has to research sensitive files of the web server.

6 Conclusion

IG is not completed, any type of program will be complete, in fact they will always need adjustments or updates, also because the technology becomes more and more complicated and more innovative. However, IG could be a first step to create automatic system that deals with cybersecurity both on the attack side and on the defensive side. Furthermore, could be interesting create a GUI of IG, maybe a browser implementing IG.

18-DBMS	NO	NO		NO	NO	NO	NO		NO
17-OS	YES	YES		YES	YES	YES	YES		YES
16-Port	YES	YES		YES	YES	YES	YES		YES
15-Traceroute	YES	YES		YES	YES	YES	YES		YES
14-Subnet	YES	YES		YES	YES	YES	YES		YES
13-Zone Transfer DNS	NO	NO		YES	NO	NO	NO		NO
12-Server Version	YES	YES		YES	YES	YES	YES		YES
11-CMS	NO	NO		NO	NO	NO	NO		NO6
10-Sub-domain	YES	YES		YES	YES	YES	YES		YES
9-DNS-Lookup	YES	YES		YES	YES	YES	YES		YES
8-Honeypot	NO	NO		NO	NO	NO	NO		NO
7-CloudFlare	NO	NO		NO	NO	NO	NO		NO6
6-Spidersite	YES	YES		YES	YES	YES	NO		YES
5-History	YES	YES		YES	YES	YES	YES		YES
4-Whois	YES	YES		YES	YES	YES	YES		YES
3-GEO-IP	YES	YES		YES	YES	YES	YES		YES
2-IP Domain	YES	YES		YES	YES	YES	YES		YES
1-Check Domain	YES	YES		YES	YES	YES	YES		YES
1-C.C CyberToscano 2-POLIMI 3-POLITO 4-UNIBA 5-UNIBO 6-UNICA 7-UNICAL 8-UNICAM 9-UNIGE									
18-DBMS	NO		NO	NO		NO		NO	NO
17-OS	YES		YES	YES		YES		YES	YES
16-Port	YES		YES	YES		YES		YES	YES
15-Traceroute	YES		YES	YES		YES		YES	YES
14-Subnet	YES		YES	YES		YES		YES	YES
13-Zone Transfer DNS	NO		NO	NO		NO		NO	NO
12-Server Version	YES		YES	YES		YES		YES	YES
11-CMS	NO		YES	NO		NO		NO	NO
10-Sub-domain	YES		YES	YES		YES		YES	YES
9-DNS-Lookup	YES		YES	YES		YES		YES	YES
8-Honeypot	NO		NO	NO		NO		NO	NO
7-CloudFlare	NO		NO	NO		NO		NO	NO
6-Spidersite	YES		YES	NO		YES		YES	YES
5-History	YES		YES	YES		YES		YES	YES
4-Whois	YES		YES	YES		YES		YES	YES
3-GEO-IP	YES		YES	YES		YES		YES	YES
2-IP Domain	YES		YES	YES		YES		YES	YES
1-Check Domain	YES		YES	YES		YES		YES	YES
10-UNILINK 11-UNINA 12-UNIPD 13-UNIPG 14-UNIFI 15-UNIROMA1 16-UNISAL 17-UNISANNIO 18-UNIVPM									

Table 1: Results of IG

References

- [1] Cis Sapienza Comitato Nazionale Ricerca in Cyber Security A Cini Cybersecurity National Lab, Cini. Cyber challenge. <https://cyberchallenge.it/>, 2018.
- [2] Andrei Broder. A taxonomy of web search. In *ACM Sigir forum*, volume 36, pages 3–10. ACM, 2002.
- [3] Meng Han and Xiao Hu Qiu. Personalized search engineer model. In *Advanced Materials Research*, volume 268, pages 1216–1221. Trans Tech Publ, 2011.
- [4] Kshitija Pol, Nita Patil, Shreya Patankar, and Chhaya Das. A survey on web content mining and extraction of structured and semistructured data. In *2008 First International Conference on Emerging Trends in Engineering and Technology*, pages 543–546. IEEE, 2008.
- [5] XiQuan Yang, DiNa Guo, XueYa Cao, and JianYuan Zhou. Research on ontology-based text clustering. In *2008 Third International Workshop on Semantic Media Adaptation and Personalization*, pages 141–146. IEEE, 2008.
- [6] Marco Spada Michele Kidane Mariam, Federico Ruzzi. The osint il mondo a portata di click. breve guida sull’intelligence da fonti aperte. In *2008 Third International Workshop on Semantic Media Adaptation and Personalization*, Scienze sociali, tecnologiche e della sicurezza, 2016.
- [7] Giuseppe Cascavilla, Filipe Beato, Andrea Burattin, Mauro Conti, and Luigi Vincenzo Mancini. Ossint-open source social network intelligence: An efficient and effective way to uncover “private” information in osn profiles. *Online Social Networks and Media*, 6:58–68, 2018.
- [8] Luigi Cristiani. Abc della sicurezza: Osint – open source intelligence. Tech Economy: <https://www.techeconomy.it/2015/10/05/abc-sicurezza-osint-open-source-intelligence/>, 2015.
- [9] Stefano Novelli. Hacklog volume 2 web hacking: Manuale sulla sicurezza informatica e hacking etico. Inforge.net; 1 edizione (24 settembre 2018), 2018.
- [10] George Kurtz Stuart McClure, Joel Scambray. Hacker 7.0. Apogeo, 2013.
- [11] Luigi Cristiani. Abc of security: Defense in depth. Tech Economy: <https://www.techeconomy.it/2015/07/14/abc-sicurezza-defense-in-depth/>, 2015.
- [12] Wolfgang Kleinwächter. Icann between technical mandate and political challenges. *Telecommunications Policy*, 24(6-7):553–563, 2000.
- [13] Diritto d’autore © 1995-2008 The FreeBSD Italian Documentation Project. Manuale di freebsd. Manuale FreeBSD: https://www.freebsd.org/doc/it_IT.ISO8859-15/books/handbook/index.html, 2013, *lastmodify*.
- [14] snitch author: Smaash year: 2015 <https://github.com/Smaash/snitch>
- [15] V1D0m author: n4xh4ck5 year: 2017 <https://github.com/n4xh4ck5/V1D0m>
- [16] censyssubdomain-finder author: christophetd year: 2017 <https://github.com/christophetd/censyssubdomain-finder>
- [17] InfoG v1.0 author: thelinuxchoice year: 2018 <https://github.com/thelinuxchoice/infog>
- [18] Th3inspector Tool author: Moham3dRiahi year: 2018 <https://github.com/Moham3dRiahi/Th3inspector>
- [19] URLExtractor author: eschultze year: 2018 <https://github.com/eschultze/URLExtractor>
- [20] BillCipher author: GitHackTools year: 2018 <https://github.com/GitHackTools/BillCipher>
- [21] infoga - gathering email information tool author: cys3c year: 2017 <https://github.com/cys3c/infoga>
- [22] 53R3N17Y author: abaykan year: 2018 <https://github.com/abaykan/53R3N17Y>