BUG BOUNTY HUNTING METHODOLOGY

Information Gathering (Recon)

- Information gathering is always the first step to be performed. Information gathering includes:
  - Collecting as much as possible of sub-domains used by the target in scope;
  - Identify the technology used within the applications (i.e. Tomcat, AngularJS, and DBMS).

Scanning

- Scanning stage is used to identify vulnerabilities based on the gathered subdomains and applications. This includes:
  - Port scanning using Nmap/Masscan, and vulnerability scanning, using for instance, BurpSuite scanner;
  - Files and directories brute-forcing, in order to identify backup files, test and development files;
  - Testing for privilege escalations, and client/server side vulnerabilities.

Exploitation

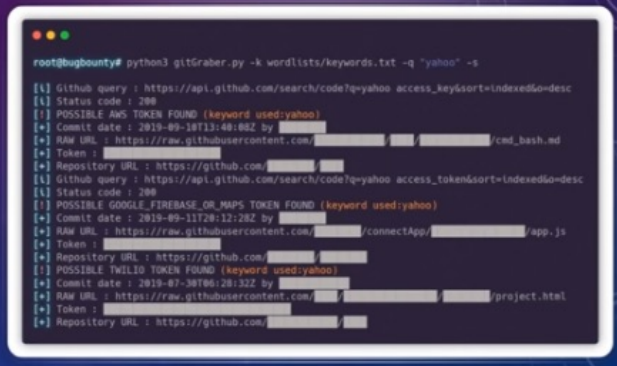
- Once a vulnerability is identified, it is now the time to try to exploit it. When exploiting a vulnerability, make sure to:
  - Never access sensitive data, but only retrieve data sample that could be used as a proof of concept;
  - Avoid abusing administrative privileges, if your vulnerability allows for privilege escalation to administrator account.

Reporting

- Once you are done with the exploitation stage, and have got a good proof of concept, it is now reporting time. Your report must include:
  - Detailed information about the identified vulnerability;
  - How an attacker could exploit it? Are certain privileges required? Or any internet user can exploit it?
  - Proof of concept (i.e. Screenshots).

INFORMATION GATHERING

- Subdomain enumeration (Sublist3r) Tool
- Sublert for domain monitoring site
- gitGraber to search for sensitive information in Github repositories
- Shodan.io
- Internet Archive (aka way back machine)
- Browser plugins: Wappalyzer, and FlagFox
- LazyRecon



SCANNING

We now have a large number of subdomains and IP's of the target, what is next?

- Nmap with –A option, and –auth –http-enum scripts (Masscan?)
- S3Scanner for insecure AWS buckets
- Subdomain TakeOver :
  - https://github.com/m4ll0k/takeover
  - https://github.com/haccer/subjack
- Run Burp scanner in the background with following plugins:
  - J2EEScan, NGINX Alias Traversal, Telewreck, ParamMiner, and Upload Scanner
- Dirsearch for backup files, hidden files, and admin interfaces
- Configure XSS Hunter with Burp Proxy (UserAgent and/or X-Forwarded-For)

Reward amounts for security vulnerabilities

New! Vulnerabilities in the Google Cloud Platform are also eligible for additional rewards under the GCP VRP Prize. The total prize money is $313,337 including a top prize of $133,337. See our announcement and the official rules for details and nominate your vulnerability write-ups for the prize here.

Rewards for qualifying bugs range from $100 to $31,337. The following table outlines the usual rewards chosen for the most common classes of bugs. To read more about our approach to vulnerability rewards you can read our Bug Hunter University article here

Category	Examples	Applications that permit taking over a Google account [1]	Other highly sensitive applications [2]	Normal Google applications	Non-integrated acquisitions and other sandboxed or lower priority applications [3]
Vulnerabilities giving direct access to Google servers					
Remote code execution	*Command injection, deserialization bugs, sandbox escapes*	$31,337	$31,337	$31,337	$1,337 - $5,000
Unrestricted file system or database access	*Unsandboxed XXE, SQL injection*	$13,337	$13,337	$13,337	$1,337 - $5,000
Logic flaw bugs leaking or bypassing significant security controls	*Direct object reference, remote user impersonation*	$13,337	$7,500	$5,000	$500
Vulnerabilities giving access to client or authenticated session of the logged-in victim					
Execute code on the client	Web: *Cross-site scripting* Mobile / Hardware: *Code execution*	$7,500	$5,000	$3,133.7	$100
Other valid security vulnerabilities	Web: *CSRF, Clickjacking* Mobile / Hardware: *Information leak, privilege escalation*	$500 - $7,500	$500 - $5,000	$500 - $3,133.7	$100

[1] For example, for web properties this includes some vulnerabilities in Google Accounts (https://accounts.google.com).

TARGET OF THE DAY

Google bug bounty program:

https://www.google.com/about/appsecurity/reward-program/

Why Google?

google -> google acquisitions
↓
Wikipidia

BUG BOUNTY INFRASTRUCTURE SETUP

Following is what I use during my recon, and actual hunting process.

1) A windows VPS with Java installed, and a good hard-disk space – Will be used to continuasly run Burp scanner
2) Two Linux VPS. Preferably running Kali. AWS provides Kali VPS – will be used to run enumeration tools, and scanners (i.e. Nmap, subdomain-takeover scanners)
3) A VPN (Optional) – in case your VPS IP got blocked or throttled, you can use the VPN to avoid/overcome such problems.

LET'S HUNT!

Lets read google bug bounty program policy page.

- Configure Burp
- Subdomain Enumeration

Note: document all possible vulnerabilities, and interesting endpoints you identify, in OneNote

350 $

CEH

Python

redhat