

**CMPE 209-03**

**Network Security**

**Spring 2017**

**Network Traffic Analyzer**

Project Proposal

Under the guidance of

**Prof. Chao-Li Tarng**

**Project Team 12**

|  |  |
| --- | --- |
| Kshama Shalini | 010763792 |
| Rohit Kotalwar | 010805080 |
| Shivang Singh | 010939851 |
| Vinay Puttanna | 010699858 |

### Overview

Cyberslacking has become a major challenge for organizations and educational institutions. It may lead to legal issues that could cause severe damage to their reputation. So far, the measures adopted to tackle it include framing of Internet usage policies and blocking of websites that pose a threat. These measures have failed to control cyberslacking completely. Because thousands of new websites are being created every day on the Internet and it has been very complicated to keep an eye on every malicious site. But it is critical to keep a check on this menace for proper functioning and sustainability of organizations.

Despite taking several precautionary measures, people tend to access resources from the Internet that causes Copyrights violation, are inappropriate or prohibited content etc. Network security field provides a useful set of tools to understand browsing behaviour of the users. They are capable of detecting visits to malicious sites by parsing the data packets and also the geographical location of such sites. Keeping in mind the fact that it is essential to control cyberslacking, in our project, we use tools such as pyGeoIP, Wireshark, Dpkt to identify the browsing patterns of the users. We are interested to find whether the users are accessing blacklisted websites and downloading content from it. If such attempts are found, we then display the geographical locations of such malicious websites on Google Maps.

### Objectives and scope

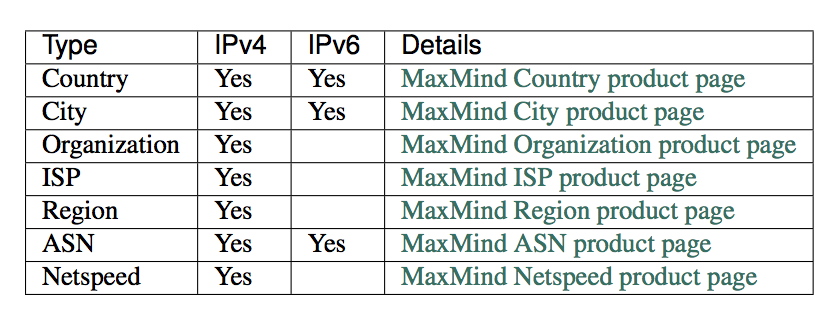
Internet is a dangerous place. Every day, around 31,000 new URLs are created that are malicious and can harm users in many ways. These malicious websites are used by hackers to attack on any system or group of systems in order to gain unauthorized control over systems and data. Finding the physical location of blacklisted websites will help the investigators in stopping and preventing these attacks.

During the course of this project, we would aim at finding the physical location of blacklisted sites. One of the main assumptions in this project is that the packets are already captured in pcap file with the help of wireshark or tcpdump. Malicious database will contain the list of all of the blacklisted sites that are malicious and can harm the users in any way. We will use scripts to parse the packets in order to find the source and destination IP addresses. Then, we will use PyGeoIP to correlate the IP address to its respective physical location. Output will be generated in CLI or KML. KML will be further mapped to Google maps for better visualization of physical locations. We will setup a proxy server to sniff usernames and passwords. We will also identify users who downloaded from any blacklisted site.

### Approaches

**pyGeoIP (IPtables)**

We will be using pyGeoIP tool for getting the Geolocation of the IP address. Python GeoIP or pyGeoIP is a python library and a python tool used to provide the geolocation information of the particular IP address. The library was developed on the top of PHP GeoIP tool.The library takes the path of filename as parameter for the creation of GeoIP class. This filename is the path to GeoIP database. GeoIP class which is created takes the IP address as parameter and returns the Geolocation information of the IP address. The Geolocation information such as country, region, city, organization, Internet Service Provider and ASN can be found out using pyGeoIP. pyGeoIP uses the following databases to access the Geolocation of the IP address.



**Wireshark**

Wireshark, an open-source network traffic analyzer is used to sniff the packets in the network. Also it will be used for network troubleshooting and for the analysis of packets.

**Dpkt Python Module**

The Python module which is used for decoding the PCAP file. By using the Dpkt module we can parse PCAP files from the Wireshark and then we can inspect each packet in the PCAP file.

**Python Scripting language**

We will be using Python scripting language to develop the application script.

### Deliverables

* A Python application (script) that parses the .pcap file and extracts source and destination IPs
* A proxy server to sniff usernames and passwords from the network traffic
* Ability to get an insight on the browsing behaviour of the network users
* Project Report conforming to the project guidelines

### References

[1] “Pure Python GeoIP API,” *Pure Python GeoIP API — pygeoip 0.3.2 documentation*. [Online]. Available: http://pygeoip.readthedocs.io/en/v0.3.2/. [Accessed: 03-Apr-2017].

[2] “How do I import a KML data file into Google Maps?,” *Ask Dave Taylor*, 15-Apr-2009. [Online]. Available: https://www.askdavetaylor.com/how\_to\_import\_xml\_kml\_data\_file\_google\_maps/. [Accessed: 03-Apr-2017].

[3] J. Oberheide, “dpkt Tutorial #2: Parsing a PCAP File,” *Jon Oberheide Full Atom*. [Online]. Available: https://jon.oberheide.org/blog/2008/10/15/dpkt-tutorial-2-parsing-a-pcap-file/. [Accessed: 03-Apr-2017].

[4] “Directory of Malicious IPs,” *Malicious IPs | By Last Bad Event | Project Honey Pot*. [Online]. Available: http://www.projecthoneypot.org/list\_of\_ips.php. [Accessed: 03-Apr-2017].

[5] http://www.huffingtonpost.com/2011/01/26/top-websites-blocked-by-businesses\_n\_814453.html