Project Team & Initial Specifications

This document will serve as an outline and high level overview of our covert C&C server and data exfiltration system.

By: **Mankirat Gulati** and **Stanley Lim**

Specifications

The objective is to exfiltrate data from an infected client to an attacker-controlled server and issue commands without detection. While there are many ways to do this, our project will utilize **DNS Tunneling**, where the covert channel is the DNS protocol.

DNS Tunneling

- In order to implement DNS Tunneling, we would make use of a domain we own and point it to our server. The server will mimic a DNS server but will embed commands in response packets. The infected client would intrepret the DNS response and execute the command. Upon execution, the client would send a receipt back to the server in a similar manner.
- **Example**: The client could send an A record request where data is encoded in the host name: MDJAEFB.z.example.com.
    - The server can answer with a CNAME response such as LAOOEFA.z.example.com.
- The data that is being transferred will be encrypted in the client and decrypted in the server to reveal its true contents and to keep communications covert.

Periodic Communication

- The server cannot directly initiate a communication with the client. As a workaround, the client can periodically send a DNS request to the C&C server and will execute a command if the server has provided one in its response.
- The encrypted command will be stored in the `Resource Data` section of the DNS response.

C&C Server

- We were thinking of using a cloud provider like AWS for our C&C server.
    - AWS usually changes the IP of a server every time it is turned off and on so we will be using Elastic IPs to ensure we can perform DNS Tunneling.
- To make our covert channel more inconspicuous, the server will be assigned a domain name that is similar to a legitimate service provided online, such as OS updates, ad servers, or services that require constant updates like weather, stocks, news, etc.

Server Payload

- The basic idea is that exfiltrated data will be communicated to the server by encrypting the data and setting it as the subdomain for the URL to our authoritative server.
- One of the main challenges is ensuring that large pieces of data transfer reliably without any leakage of data.
    - Because of the limitations placed on subdomain lengths, we need to work around a 63 char limit for each "label" in a given domain, where a label is defined as "consists of a length octet followed by that number of octets".
- A proposed solution is to take advantage of having multiple layers in our subdomain to help identify requests and store payload. To achieve this, we use one layer to store the **header** information and the other layer to store the **payload**.
    - The **header** will consist of information such as identifying the start/end of payload transferring, sequence number that determines the order of the current payload being transferred and a unique identifier for the machine (to identify and handle communication with multiple clients).
    - The **payload** consists of the data that we want to transfer.
    - A URL sent to our C&C server will follow a schema like this:

```
<header>.<payload>.domain.com
```

Infected Client

- Our infected client will be a VM running a flavor of Linux (most likely Ubuntu 16.04).

Features

- **Command Transmission**
    - Commands are transmitted to the victim machine with custom malware running to deliver any payload.

        - The malware running on the client machine will execute them after decoding the hidden string.
- **Asynchronous or Periodic Transmission**
    - Notifications of any piece of data or status on the victim machine will be transmitted periodically and asynchronously.
    - The data will return the return values of commands or return various details about the victim machine.
- **Data Exfiltration**
    - Large files and sensitive data can be transferred to and from the victim and the attacker-controlled server.
    - The data transfer may occur periodically or at random times to mask this communication.
    - Additional configurations can be included in the settings file for masking connections as being sent from other applications, like Firefox.

References

[1]. What is the maximum length of a DNS name?
[2]. DNS Tunneling w/ Iodine
[3]. Command and Control
[4]. Detecting DNS Tunneling