BCIT Comp 8505

Assignment 4

DNS Spoofing

By:

John Warren

A00087379

For:

Aman Abdulla

Due:

Nov 5, 2018 - 11am

# Table of Contents

[Table of Contents 2](#_Toc529058962)

[Introduction 3](#_Toc529058963)

[Requirements 3](#_Toc529058964)

[Objective 3](#_Toc529058965)

[Constraints 3](#_Toc529058966)

[Deliverables 3](#_Toc529058967)

[Due 3](#_Toc529058968)

[Evaluation 3](#_Toc529058969)

[Implementation 4](#_Toc529058970)

[Analysis 4](#_Toc529058971)

[ARP REQ 4](#_Toc529058972)

[ARP RSP 4](#_Toc529058973)

[DNS REQ 4](#_Toc529058974)

[DNS RSP 4](#_Toc529058975)

[Design 4](#_Toc529058976)

[Testing 4](#_Toc529058977)

[Appendix 5](#_Toc529058978)

[Source Code & Tools 5](#_Toc529058979)

[SERVER 5](#_Toc529058980)

[CLIENT 16](#_Toc529058981)

# Introduction

For this assignment you are required to implement a DNS spoofing application. This is primarily a POC application. All that is required as acceptable functionality is web site spoofing.

# Requirements

## Objective

To design a basic DNS spoofing application using any language of your choice.

## Constraints

the application(backdoor) must

with a crafted Response answer, which will direct the target system to a your own web site.

• You will test this POC on a LAN on your own systems only. This means that you are not to carry out any DNS spoofing activity on unsuspecting client systems.

• You are required to handle any arbitrary domain name string and craft a spoofed Response.

## Deliverables

Submit a zip file containing all the code and documents as described below in the sharein folder for this course under “Assignment #4”.

• Submit a complete, zipped package that includes your report, tools that you used, and any supporting data (dumps, etc), and references. Test results, complete with supporting data such as screen shots and traffic dumps

• Hand in complete and well-documented design work and documents in PDF format.

• Also provide all your source code and an executable.

• You are required to demo this assignment in the lab.

## Due

November 5, 1100 hrs.

## Evaluation

Design: 5 / 5

Documentation (explanation, user guide, etc): 5 / 5

Testing and Supporting Data: 10 / 10

Functionality: 30 / 30

Total: 50 / 50

# Implementation

## Analysis

Need to determine what machine to target – Command Line entry

Need to poison arp table to become man in the middle

Need to wait for http dns query

Grab necessary packet data

deliver DNS response

Answer as milliways

Use reference materials (arp & dns transactions) captured from lab

Reference these base packets when crafting spoof packets

### ARP REQ

A screenshot of a social media post

Description automatically generated

### ARP RSP

A screenshot of a social media post

Description automatically generated

### DNS REQ

A screenshot of a social media post

Description automatically generated

### DNS RSP

A screenshot of a social media post

Description automatically generated

## Design

### Main

parse parameters

confirm user is root

get

pthread - ARP poison gw

pthread - ARP poison target

listen for filter match

dns spoof

pthread - join (x2)

exit

### 

### ARP Poisoning

build struct

connect to socket

fill arp struct

fill ethernet struct

fill device struct

build packet

send packet(n times, f frequency, t delay)

exit

### DNS Spoofing

build struct

connect to socket

while !sigint

sniff for dns packets

grab header data

grab packet data

build header data

build packet data

send response

exit

# Testing

|  |  |  |  |
| --- | --- | --- | --- |
| # | Actions | Results | Screens |
| 1 | Capture incoming |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |

# Appendix

## References

https://www.sans.org/reading-room/whitepapers/dns/dns-spoofing-man-middle-1567

http://milliways.bcit.ca/c8505/dnsspoof.pdf

## Source Code & Captures

(attached separately)