Dale Stubbs - 14024149

Analysis of the Network Captures provided for this assignment task

Network Capture Analysis

Contents

[Abstract 2](#_Toc465690820)

[Hash Values 2](#_Toc465690821)

[Hash Table Beginning of Investigation 2](#_Toc465690822)

[Hash Table Concluding Investigation 2](#_Toc465690823)

[Capture 1 4](#_Toc465690824)

[Capture Details 4](#_Toc465690825)

[Capture Overview 4](#_Toc465690826)

[Malicious Behaviour 4](#_Toc465690827)

[Mitigation 5](#_Toc465690828)

[Capture 2 6](#_Toc465690829)

[Capture Details 6](#_Toc465690830)

[Capture Overview 6](#_Toc465690831)

[Malicious Behaviour 6](#_Toc465690832)

[Mitigation 6](#_Toc465690833)

[Capture 3 7](#_Toc465690834)

[Capture Details 7](#_Toc465690835)

[Capture Overview 7](#_Toc465690836)

[Malicious Behaviour 7](#_Toc465690837)

[Mitigation 7](#_Toc465690838)

[Appendices 8](#_Toc465690839)

[Appendix 1 8](#_Toc465690840)

[Appendix 2 8](#_Toc465690841)

[Appendix 3 8](#_Toc465690842)

[Appendix 4 9](#_Toc465690843)

[References 10](#_Toc465690844)

# Abstract

Within the following report, I will aim to analyse three different packet captures. I will aim to use all of the knowledge gained during this course to interpret the evidence presented to me and conclude if, and if so then what, any unsavoury/malicious behaviour is being perpetrated and by whom. This information will be presented and conclusion drawn based on the evidence uncovered. I will then provide some mitigation actions, which may be taken to stop malicious behaviour of this standard in the future.

# Hash Values

Hash values are unique values generated based on the contents of a specific file. If any of the information with the file is changed then the hash value will be altered greatly. This is

# Hash Table Beginning of Investigation

|  |  |  |  |
| --- | --- | --- | --- |
| File Name | MD5 Hash Value | SHA1 Hash Value | SHA512 Hash Value |
| NetworkCapture1.pcap | b834dbf7fad0d1e0  74529b03ad141110 | 050bb43df2c7e8c03ab920295370f79651733288 | 6526bfd37e7ed6e9643c9a96e49fc4e561371f292cafd8194ad7acc6adcabad6f1c9ecf3efc20871dc2e0c3e8f3d3e5004007203ecb6f5f3b87a67384f7a4178 |
| NetworkCapture2.pcap | 744d49d0487352622d624837c2cf589c | b330894d9e5243cb806cef971e478233a9d39b7a | ed66c7b49cde415ad0e4bc7abedf3ba967cdfb39318374700cef73f22e5c9364dc6e8f6b552d921fb18993a19327d622abaed421576b263d9938a1afc36ee313 |
| NetworkCapture3.pcap | e12b2d244d23b9882301cd4ad52ed1de | b02de56eb4e6aac8a6944775be95bae259cf5483 | b5cb9b447c2713833b617abe07649b895a027f46fd362e3e92fa3a5737ee2c08802d86cd8e7d1d4431800271b33cc179f22977144480917cb54f7506acd896b0 |

# Hash Table Concluding Investigation

|  |  |  |  |
| --- | --- | --- | --- |
| File Name | MD5 Hash Value | SHA1 Hash Value | SHA512 Hash Value |
| NetworkCapture1.pcap | b834dbf7fad0d1e0  74529b03ad141110 | 050bb43df2c7e8c03ab920295370f79651733288 | 6526bfd37e7ed6e9643c9a96e49fc4e561371f292cafd8194ad7acc6adcabad6f1c9ecf3efc20871dc2e0c3e8f3d3e5004007203ecb6f5f3b87a67384f7a4178 |
| NetworkCapture2.pcap | 744d49d0487352622d624837c2cf589c | b330894d9e5243cb806cef971e478233a9d39b7a | ed66c7b49cde415ad0e4bc7abedf3ba967cdfb39318374700cef73f22e5c9364dc6e8f6b552d921fb18993a19327d622abaed421576b263d9938a1afc36ee313 |
| NetworkCapture3.pcap | e12b2d244d23b9882301cd4ad52ed1de | b02de56eb4e6aac8a6944775be95bae259cf5483 | b5cb9b447c2713833b617abe07649b895a027f46fd362e3e92fa3a5737ee2c08802d86cd8e7d1d4431800271b33cc179f22977144480917cb54f7506acd896b0 |

# Capture 1

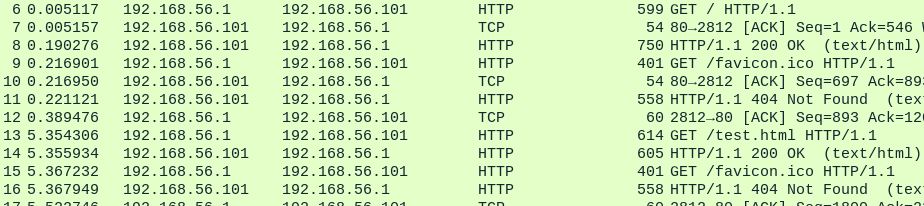
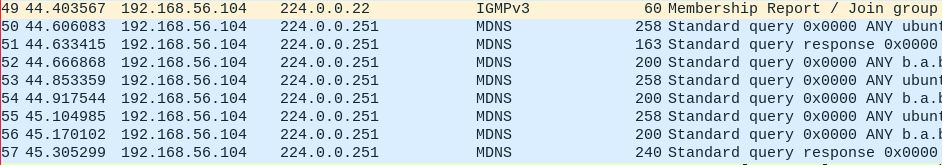
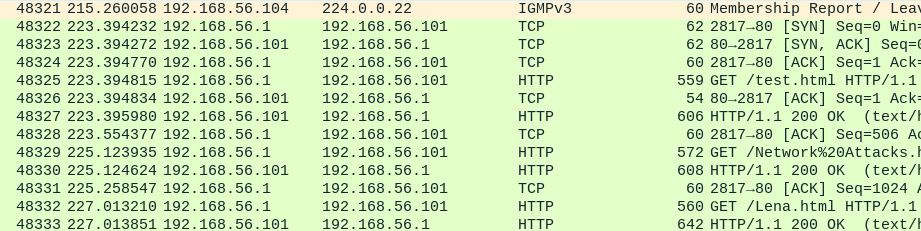
## Capture Details

This is the first Network Capture provided named ‘Networkcapture1.pcap'. My name is Dale Stubbs and my student number is 14024149.

## Capture Overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | IPv4 Address | Description | Mac Address | Role |
| Jarvis | 192.168.56.101 | Server: apache/2.2.11 | CadmusCo\_00:48:4d | Target |
| Lucille | 192.168.56.104 | N/A | CadmusCo\_1d:1b:ab | Aggressor |
| Friday | 192.168.56.1 | N/A | CadmusCo\_8e:ca:c2 | Spectator |

## Malicious Behaviour

* At approximately 16:20 on the 11/02/2010 the network capture began.
* An initialization of the TCP three-way handshake is started by Friday aimed at Jarvis using the SYN packet.
* Friday then sends out an ARP announcement to ensure that Jarvis is aware of Fridays IP and MAC Address.
* The connection between Friday and Jarvis is then completed (See appendix 1) via port 2812 of Friday and port 80 of Jarvis.
* Friday then begins to request several different items from Jarvis with little success.
* Friday then switches to port 2813 using the three-way handshake (appendix 1) and proceeds to retrieve more items from Jarvis.
* Jarvis then terminates the connections to both ports 2812 and 2813 of Friday using the FIN packet (appendix 1).
* Lucille then joins the group via IGMPv3 (appendix 2) protocol and begins a series of MDNS queries (Appendix 3).
* Lucille then sends out an ARP packet to request information about Jarvis’ IP Address.
* Lucille then pings Jarvis and then sends out and ARP packet to tell Jarvis Lucille’s’ IP and MAC Addresses
* Friday reconnects to Jarvis and requests some more HTTP Get Requests (appendix 4).
* Lucille then resends the ARP packet for Jarvis’ information.
* Lucille then begins a Sequential Port Scan on Jarvis. Packet 94 to packet 48,261 is the complete run down of the port scan. This theory is backed up by the groups of SYN requests in between single RST, ACK replies at different stages of the scan.
* At packet 48,321 Lucille leaves the group whilst Friday and Jarvis hand packets back and forth regarding HTTP requests (Appendix 4).
* Network capture concludes at 16:25 of the same date.

## Mitigation

Port scan on networks can easily be flagged up by using *‘Network security applications [can be] configured to alert administrators if they detect connection requests across a broad range of ports from a single host.’ (Lifewire, 2016)* This would enable anyone being the subject of a port scan to easily detect the port scan and can take the necessary steps to stop the attacker in their tracks.

Another way of mitigating an attack of this calibre would be to block the IP address that is guilty of port scanning you. This can be done via the router itself. You would need to have administrative privileges on the router in order to accomplish this. A fully detailed guide on how to block an IP can be found here:

<https://www.techwalla.com/articles/how-to-block-an-ip-address-from-a-network>

# Capture 2

## Capture Details

This is the second Network Capture provided named ‘Networkcapture2.pcap’. The examiners name is Dale Stubbs and my student number is 14024149.

## Capture Overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | IP Address | Description | Mac Address | Role |
| Jarvis | 192.168.56.101 | Server: apache/2.2.11 | CadmusCo\_00:48:4d | Middle man |
| Gideon | 192.168.56.102 |  | CadmusCo\_1d:1b:ab | Victim |
| Friday | 192.168.56.1 |  | CadmusCo\_8e:ca:c2 | Attacker |

## Malicious Behaviour

Dictionary password attack by Friday!!!!!!!!

Gideon is victim!!!!!

Jarvis is server being used to access Gideon’s account by Friday.

## Mitigation

# Capture 3

## Capture Details

This is the third and final Network Capture provided named ‘Networkcapture3.pcap’. The examiners name is Dale Stubbs and my student number is 14024149.

## Capture Overview

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | IP Address | Description | Mac Address | Role |
| Jarvis | 192.168.56.101 | Server: apache/2.2.11 | CadmusCo\_00:48:4d | Victim |
| Gideon | 192.168.56.102 |  | CadmusCo\_1d:1b:ab | Victim |
| Friday | 192.168.56.1 | N/A | CadmusCo\_8e:ca:c2 | Bystander |

## Malicious Behaviour

Port scan by Friday on Gideon.

Multiple SYN, ACK transmissions at port 21 – possible DoS attack?

Brute Force password attack by Friday on Gideon.

Jarvis Pings Gideon multiple times.

Jarvis and Gideon exchange HTTP requests

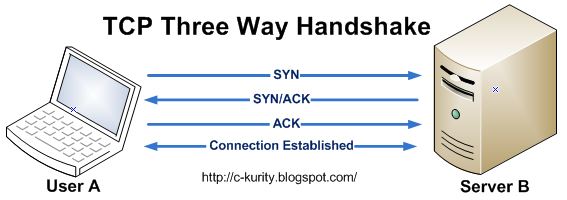
## Mitigation

## Appendices

### Appendix 1

TCP Three-Way Handshake

*“A three-way handshake is a method used in a TCP/IP network to create a connection between a local host/client and server. It is a three-step method that requires both the client and server to exchange SYN and ACK (acknowledgement) packets before actual data communication begins. A three-way handshake is also known as a TCP handshake.”* *(Techopedia.com, 2016)*



### Appendix 2

*“Internet Group Management Protocol (IGMP) and Multicast Listener Discovery (MLD) are the Multicast Group Membership Discovery (MGMD) protocols. They are essentially the same protocol, with IGMP used for IPv4 multicast groups and MLD used for IPv6 multicast groups. These protocols are used between end systems (often desktops) and the multicast router to request data for a given multicast group.”* *(Metaswitch.com, 2016)*

### Appendix 3

*“Multicast DNS is a joint effort by participants of the IETF Zero Configuration Networking (zeroconf) and DNS Extensions (dnsext) working groups. The requirements are driven by the Zeroconf working group; the implementation details are a chartered work item for the DNSEXT group. Most of the people working on mDNS are active participants of both working groups. While the requirements for Zeroconf name resolution could be met by designing an entirely new protocol, it is better to provide this functionality by making minimal changes to the current standard DNS protocol. This saves application programmers from having to learn new APIs, and saves application programmers from having to write application code two different ways — one way for large configured networks and a different way for small Zeroconf networks. It means that most current applications need no changes at all to work correctly using mDNS in a Zeroconf network. It also means that engineers do not have to learn an entirely new protocol, and current network packet capture tools can already decode and display DNS packets, so they do not have to be updated to understand new packet formats.” (Cheshire, 2016)*

### Appendix 4

*“HTTP allows for communication between a variety of hosts and clients, and supports a mixture of network configurations. To make this possible, it assumes very little about a particular system, and does not keep state between different message exchanges. This makes HTTP a stateless protocol. The communication usually takes place over TCP/IP, but any reliable transport can be used. The default port for TCP/IP is 80, but other ports can also be used. Custom headers can also be created and sent by the client. Communication between a host and a client occurs, via a request/response pair. The client initiates an HTTP request message, which is serviced through a HTTP response message in return…………*

*…………* *URLs reveal the identity of the particular host with which we want to communicate, but the action that should be performed on the host is specified via HTTP verbs. Of course, there are several actions that a client would like the host to perform. HTTP has formalized on a few that capture the essentials that are universally applicable for all kinds of applications.*

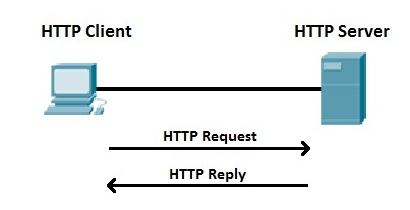
*These request verbs are:*

*GET: fetch an existing resource. The URL contains all the necessary information the server needs to locate and return the resource.*

*POST: create a new resource. POST requests usually carry a payload that specifies the data for the new resource.*

*PUT: update an existing resource. The payload may contain the updated data for the resource.*

*DELETE: delete an existing resource.” (Code Envato Tuts+, 2016)*

**

## References

Lifewire. (2016). *Wondering How Port Scanning Works? Here's the Answer*. [online] Available at: https://www.lifewire.com/introduction-to-port-scanning-2486802 [Accessed 30 Oct. 2016].

Techopedia.com. (2016). *What is a Three-Way-Handshake? - Definition from Techopedia*. [online] Available at: https://www.techopedia.com/definition/10339/three-way-handshake [Accessed 30 Oct. 2016].

Metaswitch.com. (2016). *What is Internet Group Management Protocol (IGMP) / Multicast Listener Discovery (MLD)?*. [online] Available at: http://www.metaswitch.com/resources/what-is-internet-group-management-protocol-igmp-multicast-listener-discovery-mld [Accessed 30 Oct. 2016].

Cheshire, S. (2016). Multicast DNS. [online] Multicastdns.org. Available at: http://www.multicastdns.org/ [Accessed 30 Oct. 2016].

Code Envato Tuts+. (2016). HTTP: The Protocol Every Web Developer Must Know - Part 1. [online] Available at: https://code.tutsplus.com/tutorials/http-the-protocol-every-web-developer-must-know-part-1--net-31177 [Accessed 30 Oct. 2016].