|  |
| --- |
| ci6240 – INTERNET SECURITY |
| DETERLAB PORTFOLIO |
| Daniel Carnovale |
|  |
| **K1336511** |
| **24/01/2017** |

|  |
| --- |
|  |

Contents

[Objective and Context 3](#_Toc473024195)

[Brief Background Information 3](#_Toc473024196)

[Objective of Experiment 3](#_Toc473024197)

[Experimental Method 4](#_Toc473024198)

[Blah 4](#_Toc473024199)

[Results and Discussion 5](#_Toc473024200)

[Blah 5](#_Toc473024201)

[Conclusion 6](#_Toc473024202)

SynFlood Experiment

# Objective and Context

## Brief Background Information

TCP/IP is a data transfer protocol that we use all the time when browsing the web, hence it also comes with its exploitation capabilities. It is considered “stateless”, and as the name implies there is no record or state of its actions as they are performed. This means that when the client sends a request to the server, the transmission is completely new as it has no relation or knowledge of the previous attempt. This is one of the many factors that contribute to the perpetration of certain attacks, the main types being within the Denial-of-service category (DoS). These attacks greatly affect the service the server is performing, which is the transfer of legitimate traffic towards the client. This is where the attack’s name of Denial-of-service makes sense; as the service is sent several fake packets from the attacker as if they were being sent from the client, and in a way overloads/slows down/increases the traffic resulting in a very low data transfer speed, to a point where the service is virtually unusable.

## Objective of Experiment

This experiment provides an insight into both the results of and what is carried out behind the scenes of a “TCP SYN flood” server attack. This type of attack is a DoS attack and is surprisingly easy to implement and utilise as an attacker. Because of its ease of use, the experiment also helps demonstrate methods to defend against these sorts of attacks using cookies, or in this context “SYN cookies”. The exercise will in addition elaborate on this protocol and allow observation of what happens when it’s used in practice and what vulnerabilities have been opened up when doing so.

In the context of running these experiments within Deterlab, the whole Deterlab service is dedicated to virtualising a server-client environment through uploading of an “ns” file, which will contain unique instructions for each experiment on how they should be automatically constructed in regards to the existence and placement of (in relation to each other) virtual nodes, server/client virtual computers, virtual routers, virtual hubs and so on. As a benefit of all this virtualisation, assurance is provided that the experiments are completely safe as they will not harm any physical computers. We can assume that the nodes can be treated as “points of view” from one virtual source to another. We can therefore pretend to be the server, client or attacker by connecting to these nodes in turn and performing operations such as installing the services on the server, starting a normal transfer of data from the client to the server using this server (generating traffic) or attacking the server after installing a separate service that makes use of this stream of data traffic and exploits the fact that several packets can be sent to overload it, given the stateless nature of the connection.

blah

# Experimental Method

## Blah

blah

# Results and Discussion

## Blah

blah

# Conclusion

blah