# Availability

Source: Malicious User

Stimulus: Dos Attack

Artifact: Sever

Environment: Normal run condition

Response: It should cancel the connections that the attacker has

Response measure: Number of successful requests until attack is caught

Source: Users wanting to make requests

Stimulus: A user makes a request

Artifact: Server

Environment: Normal run conditions

Response: Server handles requests in parallel

Response measure: The number of concurrent connections

# Performance

Source: A user who wants a file

Stimulus: A user requests a file

Artifact: Sever

Environment: Normal run condition

Response: The server serves the file

Response measure: The time between request and serving the content

Source: A malicious user

Stimulus: A dos attack on our server

Artifact: Sever

Environment: Normal run condition

Response: The server stops the dos attack to allow other users to not get slower performance

Response measure: Number of dos attack connections before its detected

# Security

Source: A malicious user

Stimulus: A dos attack on our server

Artifact: Sever

Environment: Normal run condition

Response: The server stops the dos attack to allow other users to not get slower performance

Response measure: Number of dos attack connections before its detected

Source: A malicious user

Stimulus: User uploads virus

Artifact: Sever

Environment: Normal run condition

Response: The server should serve the request

Response measure: The server maintains an audit trail so the managers can find the malicious user

# Tactics For Improvement

## Dos Attack – Tactic for Availability, Performance, and Security

After a threshold number of connections we ban the user from being able to make new connections and cancel the current connections. This threshold is based upon number of current connections, we allow each user to have a maximum number of concurrent connections. If a user exceeds this number of concurrent connections they will be blacklisted for a time period.

## Parallel Requests – Tactic for Availability

Tactic is to increase the capacity number of threads that can be running simultaneously on the server. This means limiting the memory impact of each individual thread. Implement the server such that it never needs to have entire files in memory or read entire contents of incoming requests before starting to service the connection. Increasing the total memory of the server could also be used for this tactic.

## Time Between Request and Content – Tactic for Performance

This tactic is to implement a caching service for the server. For the most commonly requested resources, it should be in the cache, in memory, instead of needing to be read from disk with each request.

## Audit Trail – Tactic for Security

In order to track and trace the actions of potentially malicious individuals, maintain a log of all connections, requests, and origins such that they can be looked up at a later date to determine the actions of any user on the server. This will be implemented as a log file.