

PROJECT 2: SECURITY TREE MANAGER BLACK BOX TESTS

Jake Backhouse,

CSC316: Data Structures for Computer Scientists

jabackho@ncsu.edu

North Carolina State University

Department of Computer Science

7/2/2018

BLACK BOX TEST PLAN

Testing Input Files:

Preorder File 1 (preOrder1.txt)

```
GOAL Use DDoS Attack to Disrupt All Users
O Attack Servers
O Direct BOTNET against key Servers
O 0.5 6 5000 "Rent" Existing BOTNET
O Build a BOTNET
A 0.5 3 2000 Find Vulnerable Computers
A 0.4 4 5000 Infect Computer with BOT
A 0.6 4 1000 Remain Undetected
O 0.3 7 30000 Infect Servers with Worm/Virus
O Attack Comm Infrastructure
O 0.3 7 30000 Attack Switches
O 0.3 7 30000 Attack Routers
O 0.1 8 50000 Attack DNS
O Attack All Clients
O 0.1 5 10000 Infect Clients with Worm/Virus
```

Postorder File 1 (postOrder1.txt)

```
O 0.5 6 5000 "Rent" Existing BOTNET
A 0.5 3 2000 Find Vulnerable Computers
A 0.4 4 5000 Infect Computer with BOT
A 0.6 4 1000 Remain Undetected
O Build a BOTNET
O Direct BOTNET against key Servers
O 0.3 7 30000 Infect Servers with Worm/Virus
O Attack Servers
O 0.3 7 30000 Attack Switches
O 0.3 7 30000 Attack Routers
O 0.1 8 50000 Attack DNS
O Attack Comm Infrastructure
O 0.1 5 10000 Infect Clients with Worm/Virus
O Attack All Clients
GOAL Use DDoS Attack to Disrupt All Users
```

Log Entries File 1 (logEntries1.txt)

```
9-13-2015 2:58:49,user2,save patient list
12-18-2012 16:25:58,user18,view diagnoses
7-19-2013 22:49:10,user1,edit email reminders
7-10-2015 0:26:58,user15,add patient representative list
8-1-2015 3:54:17,user10,update appointment requests
7-10-2015 6:28:13,user6,edit patient representative list
```

Log Entries File 2 (badLogFile1.txt)

```
inVALid FI1234567LE
```

Log Entries File 1 (logEntries1.txt)

900-13-2015 2:58:49,user2,save patient list

Log Entries File 2 (logEntries2.txt)

5-9-2016 0:26:58,user15,add patient to list

4-02-2008 3:54:17,user10,update requests

5-02-2009 6:28:13,user6,edit list

Test ID	Description	Expected Results	Actual Results
test01InvalidFile	Tests the loading of an invalid file. Precondition: User starts the program and indicates the attack tree pre and post files. The pre file does not exist.	User is prompted with "File Does Not Exist. Please enter new file name."	
test02RetrieveAttackGoalSummary	Tests the production of the goal (root) node of an attack tree. Precondition: User starts the program and specifies the location of the following files: preOrder1.txt and postOrder1.txt. User then chooses to see the summary of the goal (root) node of the attack tree.	The program outputs: GOAL Step[Use DDoS Attack to Disrupt All Users, C=21557.12, P=0.878, I=8.00]	GOAL Step[Use DDoS Attack to Disrupt All Users, C=21557.12, P=0.878, I=8.00]
test03RetrieveLogEntries	Tests the loading of log files that match the specified date. PreCondition: User starts the program and specifies the location of the log entries file: logEntries1.txt. User then chooses to retrieve entries based on date. User then enters the date "07-10-2015"	The program outputs: LogEntry[timestamp=2015/07/10 00:26:58, user=user15, description=add patient representative list] LogEntry[timestamp=2015/07/10 06:28:13, user=user6, description=edit patient representative list]	LogEntry[timestamp=2015/07/10 00:26:58, user=user15, description=add patient representative list] LogEntry[timestamp=2015/07/10 06:28:13, user=user6, description=edit patient representative list]
test04InvalidDate	Tests the program's ability to handle invalid date inputs. PreCondition: User starts the program and specifies the location of the log entries file: logEntries1.txt. User then chooses to retrieve entries based on date. User enters the date "09192011",	The program outputs: Invalid date. Please enter date in the format "MM-DD-YYYY". The program reprompts the user for a new date.	Invalid date. Please enter date in the format: MM-DD-YYYY Enter date in the format: MM-DD-YYYY
test05Quit	Tests the program's ability to quit properly.	The program terminates with the inputs files	The program terminates with the inputs files unedited.

	<p>Precondition: User starts the program and specifies the location of the following files: preOrder1.txt and postOrder1.txt. User then selects the quit option.</p>	unedited.	
test06Diabolical (diabolic al)	<p>Tests if the program will crash due to a totally invalid log file</p> <p>Precondition: User starts the program and specifies the location of the log file: badLogFile1.txt. If given the option, User then chooses to retrieve entries based on date. User then enters the date "01-01-2001"</p>	User is told, "Invalid File." Program stops execution.	
test07Boundary (boundary)	<p>Tests the boundary limit of dates.</p> <p>Precondition: User starts the program and specifies the location of the log file: badLogFile2.txt. If given the option, User then chooses to retrieve entries based on date. User then enters the date "12-01-2001".</p>	User is told, "Invalid File." Program stops execution.	
test08Equivalence (equivalence)	<p>Tests to make sure a log file runs the same way a different product file is run.</p> <p>Precondition: test03RetrieveLogEntries works. User starts the program and specifies the location of the log file: logEntries2.txt. User then chooses to retrieve entries based on date. User then enters the date "04-02-2008"</p>	LogEntry[timestamp=2008/04/02 3:54:17, user=user10, description=update requests]	LogEntry[timestamp=2008/04/02 3:54:17, user=user10, description=update requests]