


2019

Federal, State and Local Law Enforcement Agency Interoperability Capabilities and Cyber Vulnerabilities

Tyrone Trapnell
University of Central Florida

 Part of the [Defense and Security Studies Commons](#), and the [Information Security Commons](#)
Find similar works at: <https://stars.library.ucf.edu/etd>
University of Central Florida Libraries <http://library.ucf.edu>

This Masters Thesis (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

STARS Citation

Trapnell, Tyrone, "Federal, State and Local Law Enforcement Agency Interoperability Capabilities and Cyber Vulnerabilities" (2019). *Electronic Theses and Dissertations*. 6342.
<https://stars.library.ucf.edu/etd/6342>

FEDERAL, STATE AND LOCAL LAW ENFORCEMENT AGENCY INTEROPERABILITY
CAPABILITIES AND CYBER VULNERABILITIES

by

TYRONE N. TRAPNELL
B.S. University of Kutztown, 2007
M.S. University of Phoenix, 2015

A thesis submitted in partial fulfillment of the requirements
for the degree of Master of Science in Modeling and Simulation
in the School of Modeling, Simulation, and Training
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Spring Term
2019

Major Professor: Bruce Caulkins

© 2019 Tyrone N. Trapnell

ABSTRACT

The National Data Exchange (N-DEx) System is the central informational hub located at the Federal Bureau of Investigation (FBI). Its purpose is to provide network subscriptions to all Federal, state and local level law enforcement agencies while increasing information collaboration across all domains. The National Data Exchange users must satisfy the Advanced Permission Requirements, confirming the terms of N-DEx information use, and the Verification Requirement (verifying the completeness, timeliness, accuracy, and relevancy of N-DEx information) through coordination with the record-owning agency (Management, 2018). A network infection model is proposed to simulate the spread impact of various cyber-attacks within Federal, state and local level law enforcement networks that are linked together through the topologies merging with the National Data Exchange (N-DEx) System as the ability to manipulate the live network is limited. The model design methodology is conducted in a manner that creates a level of organization from the state level to the local level of law enforcement agencies allowing for each organizational infection probability to be calculated and entered, thus making the model very specific in nature for determining spread or outbreaks of cyber-attacks among law enforcement agencies at all levels. This research will enable future researchers to further develop a model that is capable of detecting weak points within an information structure when multiple topologies merge, allowing for more secure operations among law enforcement networks.

ACKNOWLEDGMENTS

First and foremost, I would like to extend a very much needed thank you to my wife Brooke for her support, patience, and teamwork handling our family while I completed the academic requirements for a successful Army Career. I would also like to thank my incredible thesis advisor Dr. Bruce Caulkins for your guidance, mentorship and encouragement throughout this process leading to my success. I greatly appreciate our meeting where you provided me not only insight on how to improve the model design but also how to better develop as a United States Army Officer in the years to come. I would also like to extend a thank you to the members of my thesis committee Dr. Mathew Canham, Dr. Patricia Bockelman, and Dr. Paul R. Wiegand. Dr. Canham your insight into law enforcement operations was instrumental in the development of the model and infection-based design. Dr. Bockelman your advice on the model display, document composition, and mentorship were essential in the development of the final product. Dr. Wiegand your insight into the statistical measures between infection spread increased my understanding of analysis as well as helped to develop the measured outputs. I would like to extend a thank you to Dr. Joe Kider, your cooperation with the development of the code designing the model and simulation was one of the biggest efforts within this thesis. Thank you for taking the time to explain the code in a manner that was understandable and relatable for the use within the model. I would like to extend a special thank you to Dr. J. Tomas Reynolds for providing oversight on the review and architectural design of the thesis research paper. I would also like to extend a thank you to Tiffani Marlowe for the research assistance.

TABLE OF CONTENTS

LIST OF FIGURES	viii
LIST OF TABLES	ix
LIST OF EQUATIONS	xii
LIST OF ACRONYMS/ABBREVIATIONS	xiii
CHAPTER 1: PURPOSE/INTENT	1
1.1 Introduction – The Law Enforcement Agency Network Vulnerability Model.....	1
1.1.1 Network Terminology.....	5
1.1.2 Network Topology Defined	6
1.1.3 Topology Model Designs Utilized for Simulation.....	8
1.2 Research Question I (RQI).....	9
1.3 Research Question II (RQII)	9
1.4 Research Question III (RQIII)	10
1.5 Research Question IV (RQIV).....	10
1.6 Forming a Solution	10
1.7 Thesis Organization	11
CHAPTER 2: BACKGROUND	13
2.1 Model Design, Implementation and Background Basics.....	13
2.1.1 Florida Law Enforcement Network Elements	14

2.1.2 Pennsylvania Law Enforcement Network Elements.....	18
2.2 Information Security	20
2.3 Spear Phishing	22
2.4 Infection Model Design Foundation	24
CHAPTER 3: MODEL DEVELOPMENT METHODOLOGY, DATA DEVELOPMENT, SIMULATION METHODOLOGY, AND PYTHON CODE BREAKDOWN.....	26
3.1 Model Network Development.....	26
3.2 Comma-Separated Values (CSV) File Utilization	32
3.3 Model and Simulation Data Development.....	34
3.4 Simulation Design Methodology	37
3.5 Python Code Breakdown	39
3.6 Simulation Execution and Data Collection Methodology	52
3.6.1 Statistical Data Development and Comparison	57
CHAPTER 4: RESULTS	59
4.1 Network Architecture Statistical Comparison	59
4.2 Florida Law Enforcement Network Simulation Results	61
4.3 Pennsylvania Law Enforcement Network Simulation Results	67
4.4 Federal Law Enforcement Network Simulation Results	73
4.5 Federal Web-Based Law Enforcement Network Simulation Results	78

CHAPTER 5: SUMMARY OF FINDINGS	88
5.1 Results and Contributions	88
5.1.1 Infection Rate Comparison	89
5.1.2 Research Question I (RQ1) Results	90
5.1.3 Research Question II (RQII) Results	90
5.1.4 Research Question III (RQIII) Results	91
5.1.5 Research Question IV(RQIV) Results	91
5.2 Network Vulnerability Measurement	92
5.3 Human Factor Vulnerabilities Analysis.....	94
5.4 Future Work and Limitations.....	98
APPENDIX A. HAND SKETCHED NETWORK DIAGRAM NODE LAYOUT	100
APPENDIX B. DIGITAL NODE DIAGRAMS	113
APPENDIX C. COMPLETE PTYHON CODE FOR MODEL AND SIMULATION	126
APPENDIX D. FLORIDA NETWORK DATA GRAPHS AND STATISTICS	132
APPENDIX E. PENNSYLVANIA NETWORK DATA GRAPHS AND STATISTICS.....	148
APPENDIX F. FEDERAL LEVEL NETWORK DATA GRAPHS AND STATISTICS	164
APPENDIX G. FEDERAL LEVEL WEB-BASED NETWORK DATA GRAPHS AND STATISTICS	180
REFERENCES	196

LIST OF FIGURES

Figure 1 - Drawn County and local Nodes	27
Figure 2 - regional Node Drawing	28
Figure 3 - Digital local and County Node Design	30
Figure 4 - Federal level Law Enforcement Network Structure to the state level	31
Figure 5 - Federal level Law Enforcement Network Structure to the state level	32
Figure 6 - CSV File Infection Probability Design	35
Figure 7 - Independent and Dependent Variable Breakdown.....	36
Figure 8 - Package Import Code	40
Figure 9 - Condition Setting Code	41
Figure 10 - Class Defining Code	42
Figure 11 - Top Node Development and Counter Code	43
Figure 12 - Reading CSV File Code	43
Figure 13 - state Node Development Code.....	45
Figure 14 - region Node Development Code.....	46
Figure 15 - County Node Development Code	47
Figure 16 - local Node Development Code	48
Figure 17 - Simulation Code Development	50
Figure 18 - Main Execution Code.....	52
Figure 19 - Histogram Data Set and Statistics Table.....	58

LIST OF TABLES

Table 1 - Basic Network Design Terminology	6
Table 2 - CSV File Column Data Breakdown	33
Table 3 - Simulation Infection Probability Parameter Table	38
Table 4 - Parameters and Results Output Example	54
Table 5 - Florida Network Simulation Results Data Set 1.....	62
Table 6 - Florida Network Simulation Results Data Set 2.....	62
Table 7 - Florida Network Simulation Results Data Set 3.....	63
Table 8 - Florida Network Simulation Results Data Set 4.....	64
Table 9 - Florida Network Simulation Results Data Set 5.....	64
Table 10 - Florida Network Simulation Results Data Set 6.....	65
Table 11 - Florida Network Simulation Results Data Set 7.....	65
Table 12 - Florida Network Simulation Results Data Set 8.....	66
Table 13 - Florida Network Simulation Results Data Set 9.....	66
Table 14 - Florida Network Simulation Results Data Set 10.....	67
Table 15 - Pennsylvania Network Simulation Results Data Set 1	68
Table 16 - Pennsylvania Network Simulation Results Data Set 2.....	68
Table 17 - Pennsylvania Network Simulation Results Data Set 3.....	69
Table 18 - Pennsylvania Network Simulation Results Data Set 4.....	69
Table 19 - Pennsylvania Network Simulation Results Data Set 5.....	70
Table 20 - Pennsylvania Network Simulation Results Data Set 6.....	70
Table 21 - Pennsylvania Network Simulation Results Data Set 7.....	71

Table 22 - Pennsylvania Network Simulation Results Data Set 8.....	71
Table 23 - Pennsylvania Network Simulation Results Data Set 9.....	72
Table 24 - Pennsylvania Network Simulation Results Data Set 10.....	72
Table 25 - Federal Network Simulation Results Data Set 1	73
Table 26 - Federal Network Simulation Results Data Set 2	74
Table 27 - Federal Network Simulation Results Data Set 3	74
Table 28 - Federal Network Simulation Results Data Set 4	75
Table 29 - Federal Network Simulation Results Data Set 5	75
Table 30 - Federal Network Simulation Results Data Set 6	76
Table 31 - Federal Network Simulation Results Data Set 7	76
Table 32 - Federal Network Simulation Results Data Set 8	77
Table 33 - Federal Network Simulation Results Data Set 9	77
Table 34 - Federal Network Simulation Results Data Set 10	78
Table 35 - Federal Web-Based Network Simulation Results Data Set 1	79
Table 36 - Federal Web-Based Network Simulation Results Data Set 2	79
Table 37 - Federal Web-Based Network Simulation Results Data Set 3	80
Table 38 - Federal Web-Based Network Simulation Results Data Set 4	80
Table 39 - Federal Web-Based Network Simulation Results Data Set 5	81
Table 40 - Federal Web-Based Network Simulation Results Data Set 6	81
Table 41 - Federal Web-Based Network Simulation Results Data Set 7	82
Table 42 - Federal Web-Based Network Simulation Results Data Set 8	82
Table 43 - Federal Web-Based Network Simulation Results Data Set 9	83

Table 44 - Federal Web-Based Network Simulation Results Data Set 10	84
Table 45 - Network Infection Rate Comparison.....	84
Table 46 - Severity levels of Vulnerabilities	93
Table 47- Network Security Vulnerability Assessment Criteria (modified from (Nascimento & Mesquita, 2009)	96
Table 48- Network Security Vulnerability Grading Table (modified from (Nascimento & Mesquita, 2009)	97

LIST OF EQUATIONS

<i>NI = Node Infection @ TS 500 – Initial Node Infection</i> (1)	54
<i>NIR = NI500</i> (2)	54
<i>AINI = NI4000</i> (3)	55
<i>AIR = SUMNIR4000</i> (4)	55
<i>ANI = SUMTotal Nodes Infected4000</i> (5)	55
<i>PC = N * N – 12</i> (6)	56
<i>Network Density = ACPC</i> (7)	56
<i>VHPG = 100 xVH2</i> (8)	93
<i>HFNVV = PEiΣPEi * 2.777775</i> (9)	97

LIST OF ACRONYMS/ABBREVIATIONS

AC – Actual Connections

AINI – Average Initial Node Infection

AIR – Average Infection Rate

ANI – Average Nodes Infected

CLEAN - Commonwealth Law Enforcement Assistance Network

CSV – Comma-Separated Value

CVSS - Common Vulnerability Scoring System

DHS - Department of Homeland Security

FBI – Federal Bureau of Investigation

FLEX - Florida Law Enforcement Exchange

HFNVV - Human Factor Network Vulnerability Value

II – Initial Infection

I@500 – Infection at Time Step 500

IR – Infection Rate

JNET - Pennsylvania Justice Network

KAB - Knowledge-Attitude-Behavior

NCIC - National Crime Information Center

N-DEx – National Data Exchange System

NI – Node Infection

NIR – Node Infection Rate

PC – Potential Connections

RQ – Research Question

SD – Standard Deviation

SE – Standard Error

VHP - Vulnerable Host Percentage

CHAPTER 1: PURPOSE/INTENT

This chapter provides an overview of the Law Enforcement Model Design Concept and Network Infection Spread Probability concepts that will be covered in the body of text. The aspects to be discussed will provide a broad understanding of the methodology which creates the foundation for the research conducted. The sections to be explored are network terminology, network topology defined, topology model designs utilized for simulation, and the formulated research questions. The overarching intent of this chapter as a whole is to provide a base foundation of the required background information creating a common operational picture among the law enforcement network architectures. The current cyber threat among all law enforcement agencies proves the need for a model with an informational foundation allowing easy manipulation to assess the vulnerabilities within various network structures.

1.1 Introduction – The Law Enforcement Agency Network Vulnerability Model

Law Enforcement Interoperability within the current complex operational domain creates vast challenges for informational sharing amongst all law enforcement organizations compiling increased risks with network vulnerabilities. The ability to share information across the United states at a rapid pace reduces the time that officers at all levels must wait to act. The Federal Bureau of Investigation's (FBI's) National Data Exchange (N-DEx) system provides the structural network framework to allow all agencies, from the local to federal levels, to integrate

information creating collaboration across a universal domain. The N-DEx operates in conjunction with the Federal Bureau of Investigation's (FBI's) National Criminal Information Center (NCIC) providing all available resources for criminal information to the fingertips of police officers instantly. The National Data Exchange provides over 700,000,000 searchable records from the local, state, and federal levels of law enforcement with new records added daily making the National Data Exchange a living system (NDEx, 2018). The National Criminal Information Center (NCIC) database includes 21 files (seven property files and 14-person files) which in collaboration with the National Data Exchange provides real-time support to all law enforcement agencies throughout the country.

The National Data Exchange has multiple methods of access for users which include web portal through an internet connection or by a Logical Entity Exchange Specifications (LEXS) Search/Retrieve (SR) (Management, 2018). Law Enforcement officers who choose to utilize the LEXS-SR method to access the National Data Exchange are only able to query results on their system, meaning these departments are only to pull information down from the federal level (Management, 2018). The National Data Exchange users must satisfy the Advanced Permission Requirements, confirming the terms of N-DEx information use, and the Verification Requirement (verifying the completeness, timeliness, accuracy, and relevancy of N-DEx information) through coordination with the record-owning agency (Management, 2018). The National Data Exchange (N-DEx) System utilizes a search engine improving response time, providing precision results, and improving the structured search capabilities (Management, 2018). Universal information distribution is conducted with the National Data Exchange (N-DEx) System and National Criminal Information Center (NCIC) providing a common

knowledge base for all law enforcement agencies. National Data Exchange (N-DEx) System and the National Criminal Information Center (NCIC) provide the central link for all state-level law enforcement organization to subscribe to in order to increase information collaboration creating a uniform knowledge platform.

While addressing the security of the networks the issue of interoperability remains a real and relevant issue that must be assessed in conjunction with the security to ensure future collaboration (Hawkins, 2013). Interoperability, unfortunately, does not exist between all states within the United states forcing the Federal level Law Enforcement platform to centralize information at a central location to achieve a common operational picture easily accessible across the county. The central information hub at the Federal level is the National Data Exchange (N-DEx) System which is located within the Federal Bureau of Investigation (FBI) to allow for control of the databases supplying the database for all Law Enforcement across the states (NDEx, 2018). The National Data Exchange (N-DEx) System contains the National Crime Information Center (NCIC) providing network provides online access to records concerning wanted persons, stolen vehicles, criminal histories, and other data of importance to law enforcement and criminal justice agencies (NCIC, 2000). The intersection between interoperability and network security becomes one of the most discussed issues among all law enforcement agencies within the United states creating the need to understand the complex attacks of spear phishing, the likelihood of successful intrusions and the spread of infection based on the design of the network topology (Taylor, Epper, & Tolman, 1998). Interoperability is defined as an essential communication link within public safety and public service wireless communication systems which permits units from two or more different agencies to interact with

one another and to exchange information according to a prescribed method in order to achieve predictable results by the U.S. Department of Justice (Taylor, Epper, & Tolman, 1998). The cyber security threat is a continued and persistent threat with the evolving spear phishing techniques by attackers on the law enforcement agencies. This Law Enforcement Infection Model is designed based on the effectiveness of the spear phishing attacks reflected by the Susceptibility variable built into the simulation model. Spear phishing for the purpose of this model and research design is defined as a targeting attempt to steal sensitive information identified as credentials, financial information, or personal identifiable information (PII) from a specific individual (Giandomenico, 2018). The Law Enforcement Simulation Model assumes the method of infection to be spear-phishing accounting for the infection probabilities assumed for the nodes.

The Human Factor elements within the current cyber operational domain contribute to both the levels of increased security measures as well as to the increased vulnerabilities based on the flaws regarding the human in loop. The fact of the matter is that humans by nature are lazy and individuals in today's society will eventually gravitate to the least demanding course of course (Mäses, 2015). It is inherent that mostly all security measures that are implemented require extra effort creating additional guidelines that may be bypassed, which is considered to degrade the value of security (Mäses, 2015). The associated vulnerabilities with the human in the loop have developed a situation that essentially defines the weakest link within the informational security platform as the human. In conjunction with the human vulnerability factors, there are several aspects that must be addressed to better understand the outcomes for data collection. The data collection analysis measures the vulnerabilities that are directly related

to the “Golden Rules” which are as follows: always adhering to policies, keeping passwords and pin secret, using email and the internet with care, using caution when using mobile equipment, reporting all virus thefts, and loses, and that all actions carry heavy consequences (Kruger & Kearney, June 2006). Human Factors as it pertains to the Law Enforcement Model of Network Susceptibility (LEMONS) must be calculated in conjunction with the network vulnerability to achieve the infection probability percentages to be utilized within the simulation. The measurement methods of human factors will be outlined in Chapter 2.

1.1.1 Network Terminology

Creating a shared understanding of the conceptual design to the above law enforcement model design is dependent on the universal model terminology establishing a common picture for analysis. The terms that need to be addressed are the node, edge, edge weight, and edge arrow. The node within the model is a visualization of an entity. The edge is a visual representation of a relation providing connection between two nodes. The edge weight is the available load transfer whether information or traffic flow. The edge arrow identifies the direction of travel along the edge, it is able to travel in a single direction or both directions creating various spread patterns. The following table provides an easy visual for common terms of reference in the development of the common understanding of the model development.

Table 1 - Basic Network Design Terminology

Node	Identified as a node or dot.
Edge	Identified as a line connecting two vertices (node or dot).
Directed Edge	An ordered pair of nodes that can be represented graphically as an arrow drawn between the nodes
Undirected Edge	An edge that disregards any sense of direction and treats both nodes interchangeably.
Node Degree	Based on the number of connections to nodes and the degree distribution is the probability distribution of degrees over the network as a whole.
Out Degree	The number of edges exiting a node.
In Degree	The number of edges entering a node.
Size	Is determined by the number of edges
Weight	Is directly related to the capacity of the edge in relation to the flow between nodes.

1.1.2 Network Topology Defined

The research model designed for this research thesis addresses the topology of the Federal level Law Enforcement Network down to the local level Law Enforcement Networks. The term network topology in relation to this model is defined as the pattern in which law enforcement nodes (agencies) are connected to other nodes via links (edges) (Fencl, Burget, & Bilek, 2011). The word topology descends from the Greek word *topos* which means place and *logos* which means study (Fencl, Burget, & Bilek, 2011). Four principal topologies are used in today's complex operational domain which are a bus topology, ring topology, mesh topology, and a star topology (Fencl, Burget, & Bilek, 2011). In a bus topology design is identified as a network where all the nodes are connected to singular cable or wire (Fencl, Burget, & Bilek, 2011). The ring topology design is simply a bus design in a closed network where the information travels the ring in one direction only (Fencl, Burget, & Bilek, 2011). The mesh

topology design is when all the nodes are interconnected meaning every single node is connected to all the other nodes within the network structure. The Law Enforcement Model utilizes the Star Topology method based on the local level nodes funneling through the county, regional, and state level nodes to the federal level database which would be considered the central node. The star topology method states that all the nodes must be connected to one central device, however, when clearly identifying the type of directional flow becomes relevant in the identification of the topology. In the case of this model design the direction of flow from node to node is bi-directional meaning that an infection introduced by spear phishing or malware is able to travel both ways. The star topology is considered to be the easiest method for implementation regarding the design methodology in relation to the nodes and connections within a desired network (Fencl, Burget, & Bilek, 2011). The star topology network design has advantages and disadvantages, the advantage is nodes are able to be added with ease based on the outward design (Fencl, Burget, & Bilek, 2011). The biggest disadvantage of the star topology design is the singular point of failure (Fencl, Burget, & Bilek, 2011), which in the case meaning if a local node infects the state then all the nodes under the state become at risk for infection. Star topology in the case of the Law Enforcement Model of Network Susceptibility (LEMONS) is implemented throughout the structures to reduce the probability of a network failure or relating to the current model the risk of infection from the local nodes to the federal level. Understanding the Star Topology methodology introduces increases understanding of the conceptual design of the four topology designs implemented for this simulation model.

1.1.3 Topology Model Designs Utilized for Simulation

Network Topologies implemented within a simulation model such as the Law Enforcement Model of Network Susceptibility (LEMONS) provide a foundation for comparison based on the development of the networks among all states contributing to the federal government. There are four topologies that are analyzed with the developed (fabricated) data with the intent of utilizing all topologies in conjunction with real-world data to provide network infection spread. The first topology that is considered is the Florida Law Enforcement Network based on the network information derived from the Florida Department of Justice outlined in the background section below 1.4 Florida Law Enforcement Network Elements (Chawdry, 2017) consisting of 277 total nodes. The second topology that is utilized is the Pennsylvania Law Enforcement Network based on the network information derived from the Pennsylvania Department of Justice (Department, JNET Pennsylvania Justice Network , 2018) outlined in the background section, consisting of 285 total nodes. The third topology implemented into the simulation model design is the Federal level Law Enforcement Network which only takes into consideration the Florida and Pennsylvania Networks however with the design all states could be added within the excel file databases to create a vastly complex network. An assumption is made relating to the topology of the Federal level Law Enforcement Network that the Florida Law Enforcement Network regions connect directly to the N-DEx and the Pennsylvania Law Enforcement Network regions are funneled through the central node. The assumption is being made pertaining to the central node of the Pennsylvania Network that it is the Commonwealth Law Enforcement Assistance Network (CLEAN) server located at the Pennsylvania state Police

(Department, Pennsylvania Justice Network, 2018). This topology design consists of 561 total nodes. The fourth topology model design that is analyzed is the transformation of both the Pennsylvania Law Enforcement Network and Florida Law Enforcement Network to a web-based model connected to the N-DEx in conjunction with the Florida design. This type of topology model is implemented based on the assumption within the increase in network security measures states will move toward this which allows the lower level law enforcement agencies to connect directly to the federal government database; this design consists of 405 total nodes.

1.2 Research Question I (RQI)

Does the infection spread among law enforcement nodes rapidly increase or decrease within a network, based on the probability of infection at the Federal, state, region, County and local Nodes?

1.3 Research Question II (RQII)

How does shifting the topology of law enforcement networks and increasing the susceptibility effect the rate of infection among the Federal, state, County and local Nodes?

1.4 Research Question III (RQIII)

Does modifying network topology increase network security?

1.5 Research Question IV (RQIV)

How is information flow affected with different types of network topologies?

1.6 Forming a Solution

Conducting an analysis of the research formulated on a network virus infection spread within a Law Enforcement Network advanced by (Marius Gilbert and Andrew Liebhold) in “*Comparing Methods for Measuring the Rate of Spread of Invading Populations*” set a beginning stage for analysis. The development of a Comma-Separated Values (CSV) file, nodal alignment creating network architecture within Networkx, formulated infection probabilities, and the susceptibility probability based a virus infection rate provides the foundation for manipulation within a network to determine rates of spread from various law enforcement agencies. Forming the platform for the outcome of analysis, which is the Infection Rate, is based on the conditions developed within the initial development of the model design. The conditions that are the driving factors within the model are the Susceptibly Rate, the Topology of the

Network, the Infection Probabilities of the state, region, County and local Nodes that encompass the model design. The Susceptibility condition is alternated between 0.0001, 0.001, and 0.01. The Topologies utilized forming the conditional platform are the Florida Network, the Pennsylvania Network, the Federal level Network (Florida and Pennsylvania combined), and the Federal Web-Based Network Topologies. The Infection Probability condition is assumed at the state level to be 0.001, at the region level to be 0.01, and at the county as well as local levels are randomly generated creating various conditional states. The randomly generated data will be discussed more in depth in Chapter 3 section 3.4 Simulation Design Methodology. All of the above stated conditional properties contribute to the outcome of Infection Rates based on the topology, infection probability and susceptibility to provide data for comparison as it may relate to the human factor and network security elements as discussed in chapter 2. It should be documented and understood that the original intent of the document stated above was for the analysis of the biological spread, however the simple concept for infection was manipulated to provide the layout for the various Law Enforcement Network architectures.

1.7 Thesis Organization

The organization of the following thesis research is organized in a manner which facilitates the understanding of the development of the Law Enforcement Model of Network Susceptibility (LEMONS). Section II of the thesis outlines the background information on the model development, the Florida Law Enforcement Network, the Pennsylvania Law Enforcement

Network, information security, network vulnerability, human factor vulnerability and spear phishing. Section III outlines the methods in which data was derived and the simulation was conducted to achieve measurable results. Section IV outlines the results of the simulation conducted and addresses the critical research questions pertaining to the Law Enforcement Network Infection probabilities.

CHAPTER 2: BACKGROUND

This section provides the reader with a background of the Model Design Concepts, Florida Law Enforcement Network breakdown, Pennsylvania Law Enforcement Network breakdown, Information Security, Network Vulnerability Measurement, Human Factor Vulnerability Analysis, and an understanding of Spear Phishing.

2.1 Model Design, Implementation and Background Basics

The short history of Complex Model Design utilizing python package Networkx regarding the infection of viruses on a specified network has been developed, however the ability to create a model for Law Enforcement Interoperability has not. The simulation model based on the metapopulation code developed by Timothée Poisot provided the basic code platform for node development, occupation status, and the base analysis of spread over time. The code stated above was taken and further developed to increase the usability of the simulation, CSV file manipulation, directed node output, directed connections, and directed probabilities associated directly with the County and local Law Enforcement Nodes. The code breakdown as well as the methodology derived, was taken from the Networkx Reference 1.9 (Hagberg, Schult, & Swart, 2018). The development of the CSV files utilized pertaining to the Law Enforcement Network Model works in conjunction with the python code to extract node sizes, node labels, and infection probabilities intended for infection comparison. The nature of the infection model is

based on the susceptibility of the network as a whole in conjunction with the probability of infection at the state, region, County and local levels which is assumed for the purposes of the execution and data collection. The development of the Law Enforcement Model of Network Susceptibility (LEMONS) is designed providing the ability to insert real-time infection probability data into the CSV by utilizing the human factor measurement tools alongside the network vulnerability calculation to achieve the total infection probability value. Human Factor vulnerabilities and Network Vulnerabilities create large risk areas even in the most secure networks leading to the exposure of critical or sensitive information.

Along with the network vulnerability and human factor vulnerability is the infection probability of the spear phishing attack on the desired network. Based on the structural design and size of both Florida and Pennsylvania Law Enforcement Networks, each was chosen to be utilized for the simulation analysis of network infection. The Florida and Pennsylvania Law Enforcement Networks are outlined in the following subsections based on the data collected from each of the states. The purpose for outlining each state is to ensure the understanding of the complexity of each as well as to the amount of personnel residing within a state susceptible to the human factor vulnerabilities.

2.1.1 Florida Law Enforcement Network Elements

The state of Florida geographically identifies 67 counties employing 387 law enforcement agencies. The 387 law enforcement agencies employ 46,105 sworn police officers, about 250 for every 100,000 residents. The Florida Law Enforcement network is structured with

7 regions, where each region contains a number of counties under the regional network creating multiple network domains within the state (Enforcement, 2018). The utilization of the seven regions requires a networking linkage between each region to allow for information sharing across the state which is the Florida Law Enforcement Exchange (FLEX) program (Chawdry, 2017). The seven regions use information sharing through the Florida Law Enforcement Exchange to increase the state collaboration in conjunction with the requirement to upload data to the National Data Exchange System (Chawdry, 2017). The Pensacola region is the first region containing ten counties operating under the SmartCop software platform. The Tallahassee region is the second containing thirteen counties operating under the RLEX software. The Jacksonville region is the third region containing thirteen counties operating under the LINx software platform. The Orlando region is the fourth region containing nine counties operating under the FINDER software platform. The Tampa region is the fifth region containing nine counties also operating under the COPLINK software platform. The Fort Myers region is the sixth region containing nine counties operating under the RLEX software platform, and the Miami region is the seventh region containing four counties operating under the RLEX software platform as well. The following image depicts the above regional breakdown.

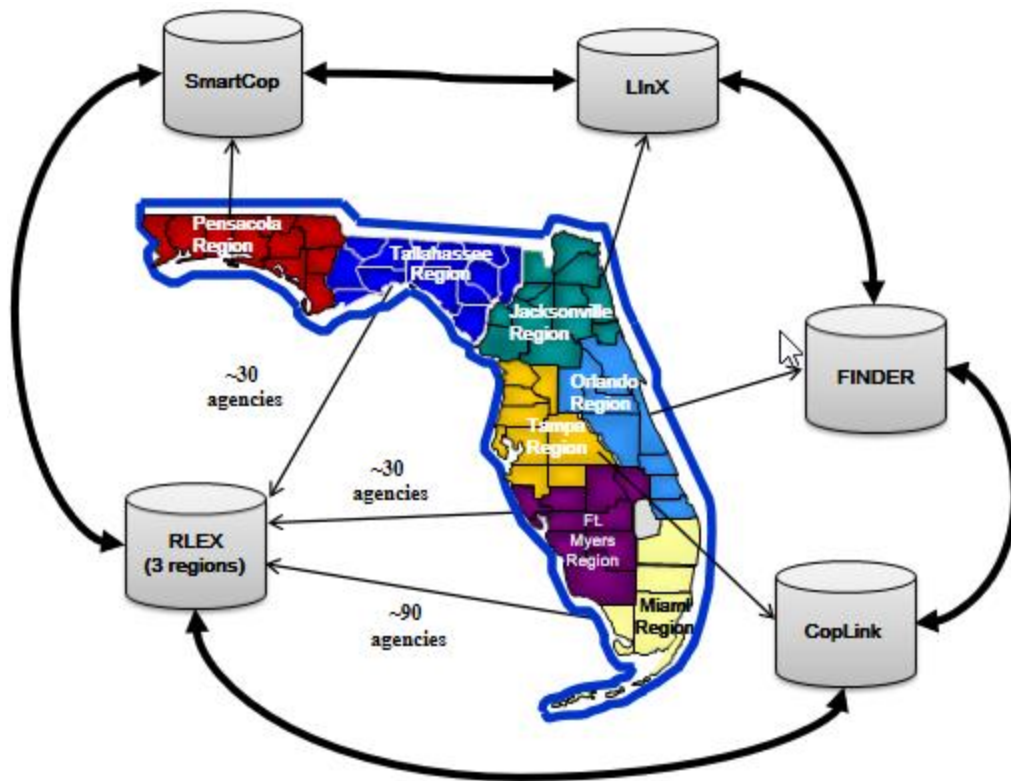


Figure 1 - Florida region Breakdown (Taken from Chawdry, R. (2017). Accelerating the Data Sharing Process. Sypher Link)

The breakdown by region identifies the implications of information sharing amongst all law enforcement agencies within the state while exposing the potential cyber vulnerabilities. The state and local law enforcement requirements for accessing informational databases provide the foundation for the operation within the identified networks. Identifying the seven regional databases converging on a neutral database connecting to the federal level database creates a level of risk based on the number of platforms and networks to be managed within the law enforcement network of Florida. Information sharing in the state of Florida within the FLEX system is defined as information that is routinely collected through the everyday operations of law enforcement agencies (Scott, 2006). Florida law enforcement agencies conduct the

information sharing operation in a manner that requires a subscription to the other networks running within the state as well as to the FLEX system network not only decreasing interoperability, but increasing the cyber vulnerabilities at various entry points (ENFORCEMENT, 2017). The state of Florida currently uses the regional hub networks from grant money distributed by the Department of Homeland Security (DHS) to provide interoperability within the state, however the state of Florida is trying to merge to the single network FLEX (ENFORCEMENT, 2017). Based on the desired network of the state the current model identified demonstrates the regional hub connecting directly to the National Data Exchange (N-DEx) System as well as the Florida FLEX which in reality creates a high level of risk for a cyber intrusion to occur. The information sharing techniques utilized by the state of Florida creates potential gaps in system connectivity requiring the need for a law enforcement agency or officer to access multiple systems at once on the issued computer platform (ENFORCEMENT, 2017). The increased requirements for cross-agency information collaboration within the state of Florida to include National Data Exchange (N-DEx) System for the purpose of state distribution exposes multiple network vulnerabilities among all agencies.

2.1.2 Pennsylvania Law Enforcement Network Elements

The state of Pennsylvania geographically identifies 67 counties which utilize 1,117 law enforcement agencies employing 27,413 sworn police officers, about 218 for every 100,000 residents. The Pennsylvania Law Enforcement network is structured with 15 regions, where each region contains a designated number of counties under the regional network point to the central network hub of the Pennsylvania Justice Network (JNET) creating a single informational domain within the state (Department, JNET Pennsylvania Justice Network , 2018). The single operational domain enables increased security management protocols, increased situational awareness throughout the entire state structure amongst all law enforcement agencies at the lowest levels. The JNET system is a web-based platform that all agency officers have access to with the appropriate credentials providing the ability to access all criminal history, search requirements, and the need to request desired information. The JNET requires a username and password login, once entering the login a verification code is sent granting access to the system for only 24 hours (Department, JNET Pennsylvania Justice Network , 2018). A new verification code is required to access the JNET system every 24 hours providing the ability to monitor access within the system. The singular platform operation creates risk management abilities identifying the cyber vulnerabilities from the local departments to the state level. region A of the JNET network contains four counties, region B of the JNET network contains four counties, region C of the JNET network contains six counties, region D of the JNET network contains four counties, region E of the JNET network contains four counties, region F of the JNET network contains nine counties, region G of the JNET network contains seven counties, region H of the

JNET network contains six counties, region J of the JNET network contains two counties, region K of the JNET network contains three counties, region L of the JNET network contains three counties, region M of the JNET network contains three counties, region N of the JNET network contains four counties, region P of the JNET network contains four counties, and region R of the JNET network contains four counties (Department, JNET Pennsylvania Justice Network , 2018). The below figure outlines the above described regions for the state of Pennsylvania.



Figure 2 - Pennsylvania region Breakdown (Taken from Department, P. J. (2018, August 28). Pennsylvania Justice Network. Pennsylvania Justice Network, United states. Retrieved from Pennsylvania Justice Network: <https://www.pajnet.pa.gov>)

The state of Pennsylvania over the last two years has operated on a server-based system connecting the entire state through servers which were housed at the county courthouses for

collaboration. The transition has begun to move to a complete web-based platform to allow every law enforcement agency to form the lowest local level to the highest federal the ability to connect and share information with a click of a button. The sharing of information is immediate with the web-based application creating the operational environment where collaboration is improving the atmosphere for the law enforcement officer during random traffic stops. The transition from the server-based network to the web-based application creates an increased risk in relation to the cyber vulnerabilities in the current cyber environment.

2.2 Information Security

Cyber vulnerabilities among many security networks lead to breaches in the confidentiality, integrity and the availability of information, thus, the ability to defend against various cyber-attacks is of utmost importance in cybersecurity. According to Von Solms and Van Niekerk (2013), information and communication technology (ICT) security describes the protection of technology-based networks where confidential information is commonly transmitted and/or stored. In cybersecurity, countermeasures or security controls are implemented to help prevent and reduce potential risks caused by cyber vulnerabilities. One of the most common forms of cyber-attacks is *phishing* – a scam, usually in the form of an email, designed to gain sensitive information from an intended victim (Workman, 2008). Attackers utilize social engineering tactics to lure and trick victims into supplying information that can be used to corrupt and/or destroy a network. In terms of information security, the assets that need

the greatest understanding and protection against cyber vulnerabilities, such as phishing, are the network infrastructure itself.

The robustness of a network infrastructure with respect to various cyber vulnerabilities, is strongly influenced by the network's topology. Often times with regard to information sharing networks, an attacker will attempt to disconnect a network or intensify the spread by destroying specific edges or nodes while a cybersecurity expert defends with resilience mechanisms (Anderson & Moore, 2006). An example of this would be an attempt to shut down an organization's file-sharing network, like that of a Florida's or Pennsylvania's law enforcement network. (Albert, Jeong, and Barabási (2000)) discovered that specific real-world networks with scale-free degree distributions are less susceptible to random cyber-attacks than to targeted attacks. This is related to how the topology of many scale-free networks depends on a minority of nodes with high vertex orders to gain connectivity, however, this does not mean their impenetrable. If an attacker was capable of destroying the "central" nodes within the network, then all connectivity and security would be lost.

Understanding how state and local law enforcement networks are structured provides insight into the potential cyber-attacks that the system may be susceptible to. However, exactly how these cyber-attacks spread throughout law enforcement networks is still unknown, therefore the ability to stop damage from spreading is limited. This study aims to introduce a model that will simulate how various cyber-attacks spread and defense measures interact.

2.3 Spear Phishing

The most common attack and infection occurring within the rapidly changing cyber operational domain is Spear Phishing not only on individuals with no connection but increasing by the day against government agencies. Extensive research has been conducted to examine the effectiveness of phishing attacks and defenses, based on the research paper “User Context: An Explanatory Variable in Phishing Susceptibility” written by Kristen K. Greene, Michelle P. Steves, Mary F. Theofanos, and Jennifer Kostick (Greene, Kostick, Steves, & Theofanos, 2018). The problem of spear phishing is not solved but rather it is an increasing issue within the cyber community that all law enforcement organizations are facing (Greene, Kostick, Steves, & Theofanos, 2018). It is critical to understand that spear phishing has evolved over time shifting from the traditional scam of obtaining user name and password to the implanted viruses within emails (Greene, Kostick, Steves, & Theofanos, 2018). Spear phishing is determined to be successful not based solely on the success of system vulnerabilities but rather human deception based on the human factors aspect addressed above (Greene, Kostick, Steves, & Theofanos, 2018). The ability to understand and recognize spear phishing emails many organizations developed training programs for individual within the specified organize to reduce the human factor, providing the ability to address the consequences associated with the spear phishing attacks (Greene, Kostick, Steves, & Theofanos, 2018). The research was conducted within a lab over a 4 ½ year time frame to collect the data acquiring the results relevant to the Law Enforcement Model Design. Along with the lab exercise, there was a survey that accompanied examining non-lab participants assessing their responses regarding the spear phishing attempts

made on them (Greene, Kostick, Steves, & Theofanos, 2018). The results for the experiment are broken down into three categories which are receiving a digital voicemail, receiving an unpaid invoice, and an order confirmation (Greene, Kostick, Steves, & Theofanos, 2018). The results for the new digital voicemail spear phishing instance are that 11.6 percent rate (8 out of 69) clicked on the voicemail, and 21.3 percent (13 out of 61) of the surveyors did not click on the voicemail. The results for the unpaid invoice spear phishing instance are that 20.5 percent rate (15 out of 73) clicked on the invoice, and 25.9 percent (15 out of 58) of the surveyors did not click on the invoice. The results for the order confirmation spear phishing instance are that 9.1 percent rate (6 out of 66 subjects tested) clicked on the order confirmation, and 50.0 percent (30 out of 60 subjects tested) of the surveyors did not click on the order confirmation (Greene, Kostick, Steves, & Theofanos, 2018). This research was then able to be taken to develop an overall assessment of the percentage of vulnerability to accepting the spear phishing which was assessed to be between the range of 43.8 percent to 49.3 percent of users accepting spear phishing attacks (Greene, Kostick, Steves, & Theofanos, 2018). The assessment of the spear phishing infection probability for the Law Enforcement Model Design is an assumption that determines the level of susceptibility of the networks in relation to identifying the attacks and stopping them before reaching the node levels. This research could be further developed to assess the vulnerability of the network in relation to the detection of spear phishing attempts reaching their target locations.

2.4 Infection Model Design Foundation

Infection based model designs have been used in various cases to determine human virus infection patterns as well as spread rate based on selected nodes within a specific model design. The first case scenario that was analyzed to understand a simple design utilizes the SIR model, which is a model based on the susceptibility, infection and removal of a virus within the human loop of infection (Shulgin, Stone, & Agur, 1998). The development of this model-based code does create a platform for understanding the Networkx package within python, however for the purpose of the Law Enforcement Model of Network Susceptibility (LEMONS) the removal of the infected nodes does not apply to the real-world application. The rational reasoning for this is the nodes represent the law enforcement agencies within the states and as they become infected, they still have a real-world mission to conduct and must be neutralized within the infected network. The ability to become un-infected within the model is a critical piece that will be addressed within chapter 5. Although there are many cases found that utilize the SIR model platform the constructs were not suitable for the development of the Law Enforcement Model of Network Susceptibility (LEMONS). The second model analyzed was the occupancy infection model design created by Timothée Poisot referencing the base python code within the “*Using Networkx to Simulate Metapopulations in Python*” which utilizes a population of nodes being occupied at over several iterations. This particular case of code that was formed determined the occupancy based on the probabilities associated with each node comparing them to random numbers determining the infection status at a random rate. The research and development conduct by Timothée Poisot were instrumental in the development of the basic model code for

the Law Enforcement Model of Network Susceptibility (LEMONS). The concept of the code provided the ability to understand the random number comparison determining the infection status amongst the nodes as well as the spreading rate throughout the model. The framework as well as foundation created by the research of these past models enabled the development of the Law Enforcement Model of Network Susceptibility (LEMONS), needed to replicate the live network for purposes of infection spread probabilities. The background research conducted by various network models such as the two described above provided the mold for the Law Enforcement Model of Network Susceptibility (LEMONS) enabling modification using CSV files to build the model. The past research and development of models within python have been conducted based on the instructions derived from the Networkx reference (Hagberg, Schult, & Swart, 2018) and the expert experience of Dr. Joe Kider. The development of the model utilizing the CSV file is instrumental in the data extraction for the purposes of the node placement as well as the size and infection probabilities for execution within the design. Although the Networkx reference guide clearly identifies the code for utilizing the CSV file for data input there is no direct reference for utilization leaving the ability for usage open for interpretation on implementation.

CHAPTER 3: MODEL DEVELOPMENT METHODOLOGY, DATA DEVELOPMENT, SIMULATION METHODOLOGY, AND PYTHON CODE BREAKDOWN

This section describes the basic model developmental methods used in relation to the past, current, and potential future network architectures, the basic infection probability data development, simulation methods in detail, and a breakdown of the python code utilized for simulation execution. The model developed in this section is intended for users who want to replicate or build off the model in order to increase network architecture foundation. The approach utilized in the following section utilized randomly generated data sets capable of being replaced with real-time data of Law Enforcement Agency Network Statistics. The Python Code approach that was used is outlined in detail within this section to provide the novice coder the ability to replicate the model and simulation providing a very understandable foundation for coding purposes ensuring there is a common operational picture developed.

3.1 Model Network Development

The interoperability capabilities among the Federal, state, County and local law enforcement agencies present a cyber vulnerability that is not afforded the appropriate attention regarding the current technological era within the Law Enforcement Community. The initial design concept of the base model was developed utilizing paper and pencil to illustrate the node design. The purpose of diagraming the node concept on paper developed the foundation for the

quantity of nodes created at the local level without degrading the model design. Creating the model layout on paper enabled the analysis of the Florida and Pennsylvania Law Enforcement Networks from the region level to the county level and down to the local levels. Only three local level nodes were utilized in order to model the infection rate as well as information flow, while at the same time maintaining the ability to keep the model size reasonable. The initial step before drawing out the nodes was to determine the branches of state nodes from the FBI N-DEx system to the state level nodes. The next step was determining the number of regions identified within each state which created the foundation for the number of sub system networks connecting the state levels or the FBI N-DEx system. It is critical to understand the amount of time and work expended analyzing the sub networks of each state, which required identification and mapping of the counties as well as local nodes for each specific regional level node. The following figure displays the initial county and local level node mapping on paper:

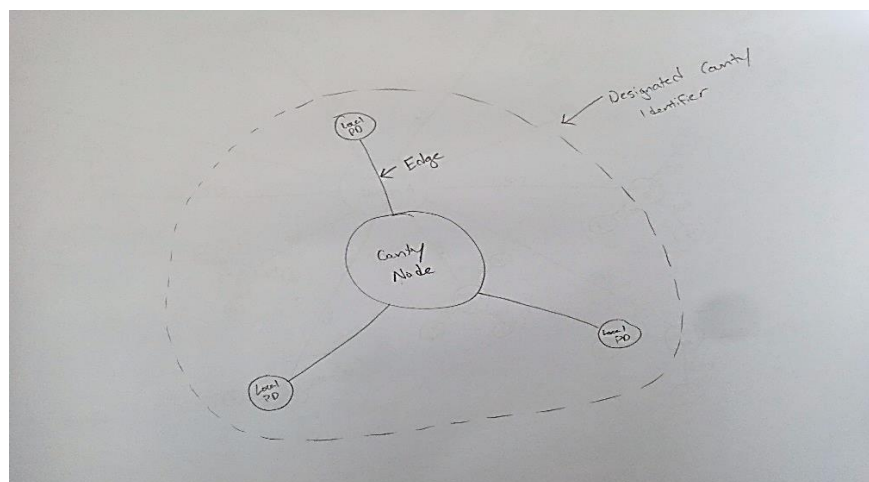


Figure 3 - Drawn County and local Nodes

After completion of the county and local level node layout the next objective is to outline the number of counties per region on paper providing the visualization for each regional level. The importance of this step for each region is the connections between the region to the county down to the local level nodes. The connections (edges) between the regions and counties are critical in the aspect of connection identification for the future development of the relationships among all the nodes. The following figure is an example of a Florida Network regional node drawn out with the respective number of counties and three local nodes per county node.

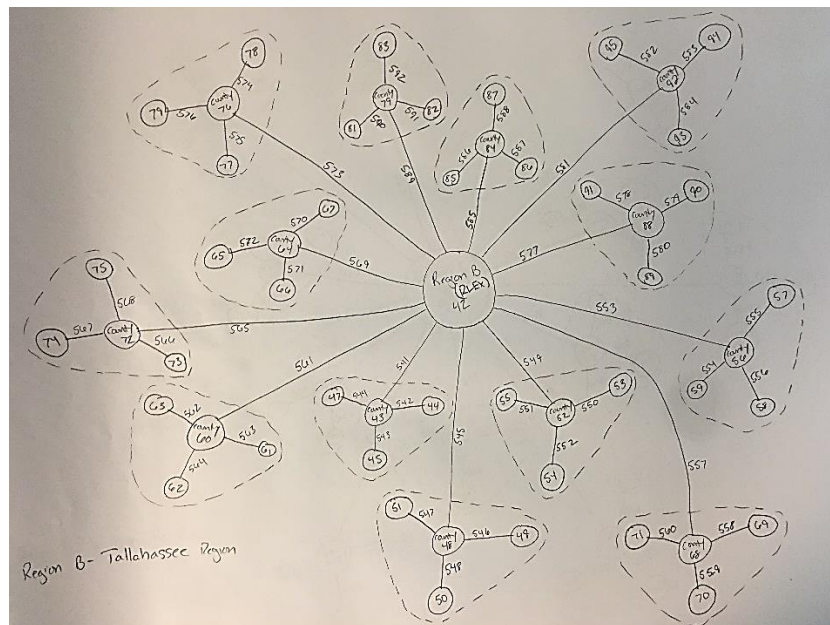


Figure 4 - regional Node Drawing

Completion of the regional node drawing outlines with the respective counties and local nodes for both of the states (Florida and Pennsylvania) develops the operational platform for the state composition network architecture. Once the regions are detailed on paper the following

design is to map the Florida regions to the N-DEx and the Pennsylvania regions through the JNET node to the N-DEx Node. This provides the overarching outline of the regions to the states to the FBI database essentially creating the top-level architecture of the model. The drawing outlining the regions to the states to the FBI N-DEx system node can be referenced in Appendix A in conjunction with all preliminary drawings developing the initial paper drawn model providing the foundation for the Law Enforcement Model of Network Susceptibility (LEMONS) development. Accompanying the initial drawings for understanding the operational platform is a digital set of prints were developed for the ability to more clearly visualize the connections and relationship among all nodes. This set of digital prints is able to be referenced in Appendix B for a visual representation of the networks initially developed.

Utilizing the above referenced drawings of the node relationships, a basic model design is developed using Python code in collaboration with the Networkx package providing the ability to replicate the layout of the various network structures among law enforcement agencies. The python code derived was utilized in collaboration with the research of Timothée Poisot referencing the base python code within the “*Using Networkx to Simulate Metapopulations in Python*” resource which can be found at <http://timotheepoisot.fr/2012/05/18/networkx-metapopulations-python>. The code used within the Law Enforcement Model of Network Susceptibility (LEMONS) will be outlined in further detail later in this section. For the purpose of the Law Enforcement Model of Network Susceptibility (LEMONS) design the states of Florida and Pennsylvania law enforcement networks are used based on the network structure of the past and current network architectures. The states of Florida and Pennsylvania are replicated utilizing the node and edge concept to design the network structure providing the replication of

the flow of network traffic from the lowest local level law enforcement agency to the National Data Exchange (N-DEx) system located within the FBI. Based on the number of local law enforcement agencies within various counties to the state level, the decision to replicate and simulate only three agencies at the local level increases the scope of the model development referenced in the figure 3 below.

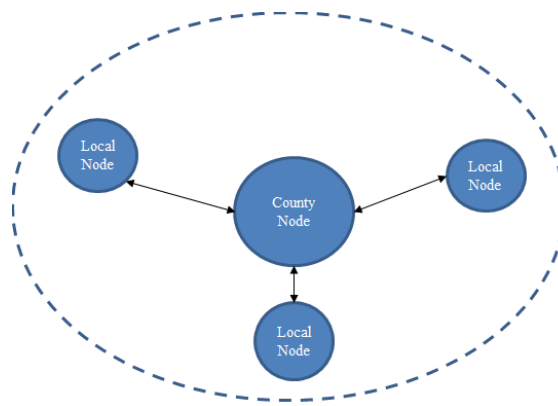


Figure 5 - Digital local and County Node Design

Utilizing small scale node composition provides the ability to deep dive into local agencies while at the same time keeping the broadness of scope open for a large-scale simulation. Assigning only three local level nodes to the county nodes increases the state participation within the model design increasing the various platforms from all states to be incorporated within the model. For the purpose of this particular model, the states of Florida and Pennsylvania are developed independently from each other to examine the flow of information based on the state-dependent architecture. Florida Law Enforcement Network structure contains seven regional nodes and the state of Pennsylvania contains fifteen regional nodes. The

overarching network architecture demonstrated in figure 4 displays the connection from regional nodes to state level to the National Data Exchange (N-DEx) System.

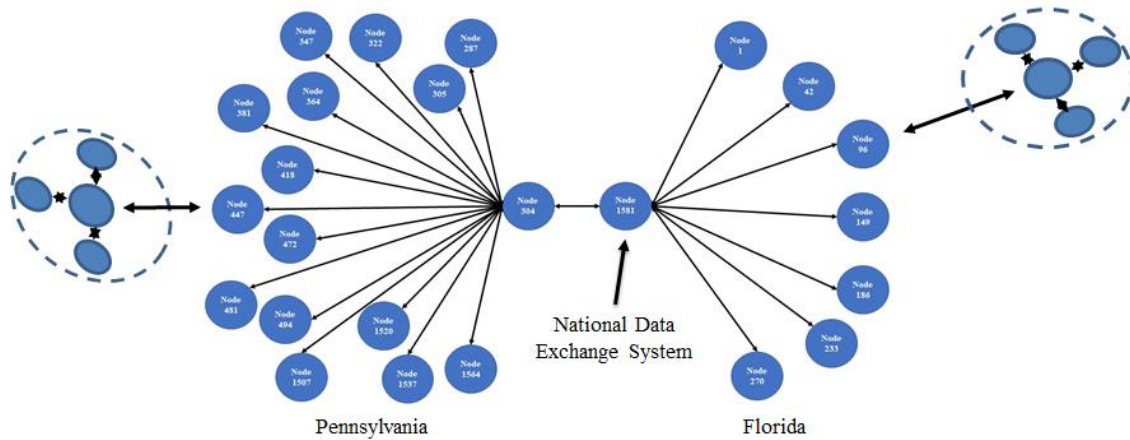


Figure 6 - Federal level Law Enforcement Network Structure to the state level

After completion of the replication of the past and current network structure utilized in both states with various alterations in the node levels, the analysis was conducted identifying the shift in technology creating the need to shift the network architecture. Examining the Florida Department of Justice (Enforcement, 2018) and the Pennsylvania Department of Justice (Department, Pennsylvania Justice Network, 2018) the manipulation regarding the topology of the network structure is able to be done by eliminating the region level and county level nodes within the python code creating the web-based network that many states are moving towards based on the technological shift. Conducting this manipulation in the network architecture produces a web-based structure for both states connecting to the federal level National Data Exchange System referenced in the figure 5 below.

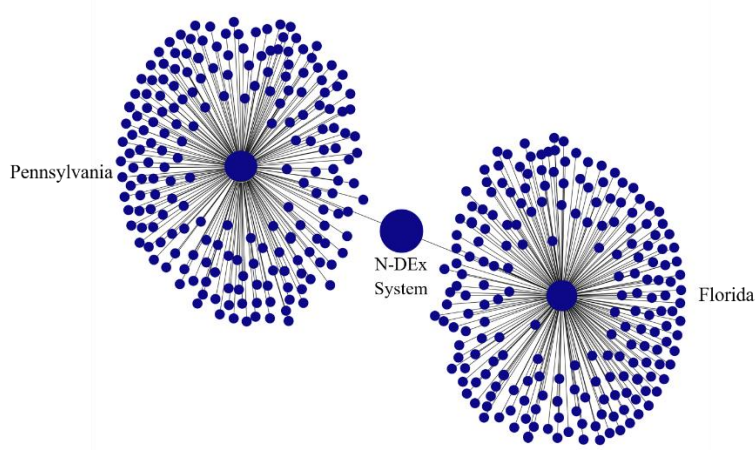


Figure 7 - Federal level Law Enforcement Network Structure to the state level

3.2 Comma-Separated Values (CSV) File Utilization

The Law Enforcement Model of Network Susceptibility (LEMONS) structure design is driven from the use of a Comma-Separated Values (CSV) file(s), identifying the variables (state, state node size, state infection, region, region node size, region infection, county, county node size, county infection, local, local node size and local infection) utilized to build the model as well as conduct the simulation. Understanding the data input from each column with the CSV file is critical in the development and execution of the simulation iterations allowing for random data creation. The state column identifies the state level node; the state node size identifies the numerical size of the state node; the state infection column inputs the infection probability of the state node; the region column identifies the region level node; the region node size identifies the numerical size of the region node; the region infection column inputs the infection probability of

the region node; the county column identifies the County level node; the county node size identifies the numerical size of the county node; the county infection column inputs the infection probability of the county node; the local node size identifies the numerical size of the local node; the local infection column inputs the infection probability of the local node; this information database is able to be manipulated based on the state network procedures. Data breakdown of the CSV file columns are organized in specific column ordered format creating the database foundation for both the model structure as well as the simulation execution. The four CSV files that are used for databases are the Florida Law Enforcement Network, Pennsylvania Law Enforcement Network, Federal Law Enforcement Network and the Federal Web-Based Law Enforcement Network. The purpose for the four different files is designed for the ability to input accurate real time data into the database in order to enhance the simulation network replication. The following table identifies the breakdown of the CSV file in conjunction with the data inputs required for the model structure and simulation execution.

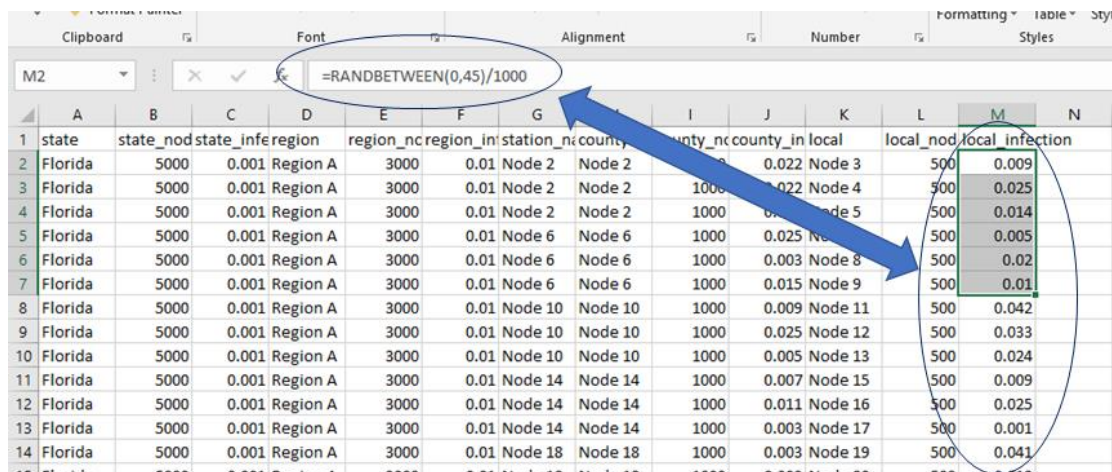
Table 2 - CSV File Column Data Breakdown

state	state_node_size	state_infection	region	region_node_size	region_infection	station_name	county	county_node_size	county_infection	local	local_node_size	local_infection
Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0.16	Node 3	500	0.02
Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0.03	Node 4	500	0.11
Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0	Node 5	500	0.16
Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.15	Node 7	500	0.16
Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.17	Node 8	500	0.07
Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.24	Node 9	500	0.2
Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.2	Node 11	500	0.38
Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.17	Node 12	500	0.45
Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.1	Node 13	500	0.19
Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.11	Node 15	500	0.5
Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.25	Node 16	500	0.1
Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.22	Node 17	500	0.52
Florida	5000	0.001	Region A	3000	0.01	Node 18	Node 18	1000	0.06	Node 19	500	0.72
Florida	5000	0.001	Region A	3000	0.01	Node 18	Node 18	1000	0.16	Node 20	500	0.65

3.3 Model and Simulation Data Development

The node size data development is dependent on the relation of the node to neighboring nodes within model providing identification of the nodes. The Top-level node is identified as the National Data Exchange (N-DEx) System, which is set at size 10,000. The state level nodes are identified as the states of Florida and Pennsylvania, which are set at size 5000. All region level nodes identified within the model are set to node size 3000. All County level nodes identified within the model are set at node size 1000. All local level nodes identified within the model are set at node size 500. Federal, state, County, and local law enforcement agencies cyber protocols and databases are considered to be sensitive information in nature creating potential risk in the unclassified model development. For the purpose of the model design described above the informational database containing the infection probabilities in relation to the state, region, County and local nodes the probabilities were developed at random (fabricated). The following numerical values are all assumed. The state level node for the model is assumed to be set as 0.001 (.01%) and the region level node is set at 0.01 (1.0%). The county and local nodes were derived utilizing the Random Number function within excel creating an unbiased range of probabilities between the desired range. The county level infection probability utilized four levels of probabilities which are as follows: =RANDBETWEEN (0,2)/100, =RANDBETWEEN (0,15)/100, =RANDBETWEEN (0,25)/100, and a fixed probability set at 0.0001. The local infection probabilities utilized four levels of probabilities which are as follows: =RANDBETWEEN (0,5)/100, =RANDBETWEEN (0,25)/100, and =RANDBETWEEN (0,75)/100. The final data set utilized in the simulation execution is where the state is equal to 0.0001, the region is set to 0.001, the county level is set 0.0001, and the local levels are set to

0.0001 (only 1 node starts as infected). The use of the ranges for the both the county and local levels imply security measures have been conducted associating the levels of risk with the randomly generated probabilities which either identifies the risk probabilities of the law enforcement agency. The use of the randomly generated data creates the ability to input various data sets into the model and simulation. The development of the database for each model enables real time data to replace the randomly generated data creating a real time model for real data analysis. Development of real data which was outlined in Chapter 2. The above referenced development method is outlined in figure 6 below.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	state	state_nod	state_inf	region	region_nc	region_in	station_nc	county	county_nc	county_in	local	local_nod	local_infection	
2	Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0.022	Node 3	500	0.009	
3	Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0.022	Node 4	500	0.025	
4	Florida	5000	0.001	Region A	3000	0.01	Node 2	Node 2	1000	0.025	Node 5	500	0.014	
5	Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.025	Node 6	500	0.005	
6	Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.003	Node 8	500	0.02	
7	Florida	5000	0.001	Region A	3000	0.01	Node 6	Node 6	1000	0.015	Node 9	500	0.01	
8	Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.009	Node 11	500	0.042	
9	Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.025	Node 12	500	0.033	
10	Florida	5000	0.001	Region A	3000	0.01	Node 10	Node 10	1000	0.005	Node 13	500	0.024	
11	Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.007	Node 15	500	0.009	
12	Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.011	Node 16	500	0.025	
13	Florida	5000	0.001	Region A	3000	0.01	Node 14	Node 14	1000	0.003	Node 17	500	0.001	
14	Florida	5000	0.001	Region A	3000	0.01	Node 18	Node 18	1000	0.003	Node 19	500	0.041	

Figure 8 - CSV File Infection Probability Design

The establishment of the infection probability data is only one piece of the puzzle needed in relation to the development of the desired outcome. The Law Enforcement Simulation Model is designed utilizing 3 independent variables which contribute to the direct output of the direct variable. The three Independent variables within the model are the Susceptibility of the network, the topology of the network and the infection probabilities of the nodes. The infection

probabilities consist of the data outlined in the above paragraph. These three variables directly produce the infection rate (dependent variable) through the simulation of the data. Upon the output of the dependent variable (infection rate), creates the ability to calculate statistical data for comparison among the various data sets in relation to the network architectures selected as defined in table 6. The statistical data outputs that are utilized are as follows: the mean, the standard error, the standard deviation, the sample variance, Kurtosis, and skewness. The following figure clearly demonstrates the independent variables feeding the dependent variable as well as the statistical outputs.

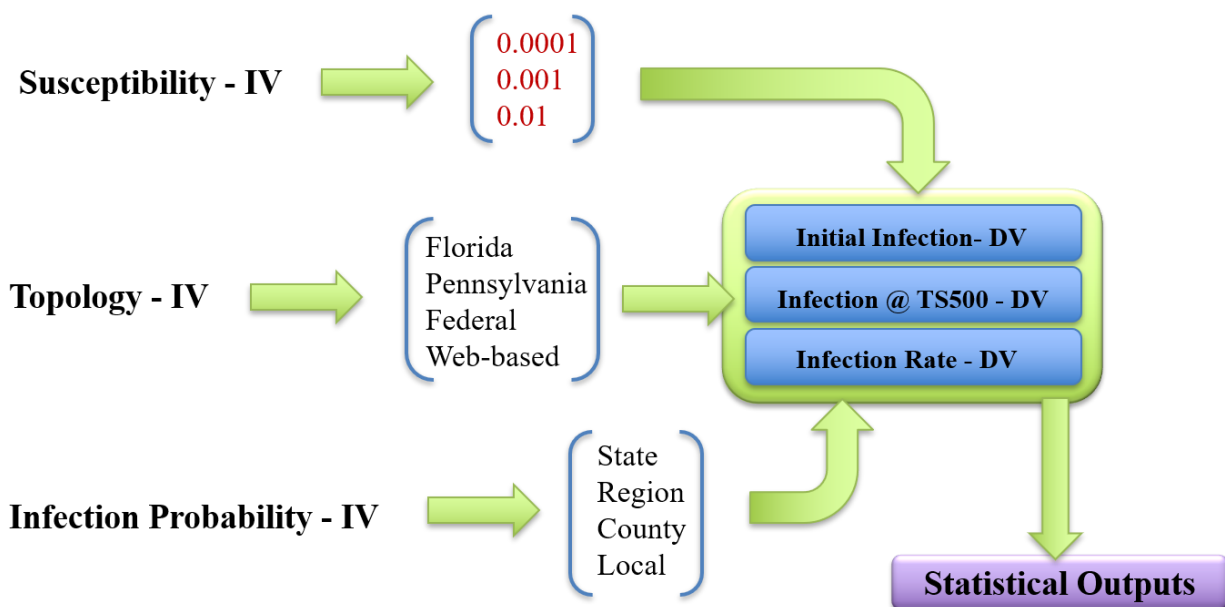


Figure 9 - Independent and Dependent Variable Breakdown

3.4 Simulation Design Methodology

The purpose of the simulation model design is to replicate cyber intrusions at the local, county, state and federal levels simulating the rate of spread of an infecting virus introduced to the network. The introduction of a virus is dependent on the probability of risk-oriented behaviors directly relating to the probability designated to each node representing the susceptibility of infection directly impacting the rate and directional spread within the model. The probability is then compared against the random function within python providing a base of random choice numbers introducing the ability to randomly identify infected nodes. The simulation model is designed in such a manner that the National Data Exchange (N-DEx) System is represented by Node 1581, the regions of the Florida Law Enforcement are identified with the respective nodes 1, 42, 96, 149, 186, 233, and 270. The Pennsylvania Law Enforcement Network funnels through the JNET (CLEAN SERVER) which is identified as Node 304 establishing the connection from the state of Pennsylvania to the National Data Exchange (N-DEx) System. The simulation is designed to run one iteration at 500 timesteps to provide 500 possible times of infection over the network. There were nine total simulation models that were developed to dissect the infection spread through the law enforcement networks to the lowest nodes. Before the 500 timesteps are conducted the simulation starts by producing an initially infected plot identifying the initial nodes within the simulation based on the infection probabilities. The methodology of simulation development was derived from the article *Comparing Methods for Measuring the Rate of Spread of Invading Populations* where there were certain parameter aspects maintained to ensure simplicity and unbiased measurements. In the

Florida, Pennsylvania, Federal level, and Federal Web-Based level Law Enforcement Network Simulations the parameters that were maintained are probability of infection, probability of initial infection, and edge weight. The probability of infection is the measurement of 0.0001 compared to the status of the node which is determined with the formula $\text{np.random.uniform}() < \text{Infected}$, which is comparing the random number to the probability of infection on a specific node to determine the infection status. The probability of initial infection is the measurement of 0.001 compared to the status of the first node (1581), which is determined by the formula $1 \text{ if } \text{np.random.uniform}() < \text{Initial_Infection} \text{ else } 0$. This means if the random number generated is less than 0.001 then the node 1581 is infected; else 0 means the node is not infected. The edge weight which is maintained at the measurement of 0.005 which determined the flow of information within the undirected graph. These simulation parameters are outlined in the table 6:

Table 3 - Simulation Infection Probability Parameter Table

Simulation Run Data Sets	Susceptibility	State Infection	Region Infection	County Infection	Local Infection
1	0.0001	0.001	0.01	RANDBETWEEN(0,2)/100	RANDBETWEEN(0,5)/100
2	0.001	0.001	0.01	RANDBETWEEN(0,2)/100	RANDBETWEEN(0,5)/100
3	0.01	0.001	0.01	RANDBETWEEN(0,2)/100	RANDBETWEEN(0,5)/100
4	0.0001	0.001	0.01	RANDBETWEEN(0,15)/100	RANDBETWEEN(0,25)/100
5	0.001	0.001	0.01	RANDBETWEEN(0,15)/100	RANDBETWEEN(0,25)/100
6	0.01	0.001	0.01	RANDBETWEEN(0,15)/100	RANDBETWEEN(0,25)/100
7	0.0001	0.001	0.01	RANDBETWEEN(0,25)/100	RANDBETWEEN(0,75)/100
8	0.001	0.001	0.01	RANDBETWEEN(0,25)/100	RANDBETWEEN(0,75)/100
9	0.01	0.001	0.01	RANDBETWEEN(0,25)/100	RANDBETWEEN(0,75)/100
10	0.0001	0.001	0.001	0.0001	0.0001(1 Node Infected)

Utilizing the infection parameter data sets from the above table to execute the simulations of Florida, Pennsylvania, the Federal level Network and the Federal Web-Based Law enforcement Network, provides the ability to achieve quantifiable data for comparison and analysis each data set was run 4000 iterations at 500 timesteps. The purpose of the simulation runs of 4000 iterations created the ability to derive an average for the statistical measures of

initial node infection, average infection rate, and the average total node infection. The execution of one simulation run takes an average of about 1 second to execute, record the data, and 1 second to calculate the node infection at timestep desired. The execution of the 4000 iterations is achieved with a Run.py file created to execute the script 4000 times while simultaneously recording the data into a text file analysis for recoding and calculation purposes.

3.5 Python Code Breakdown

The purpose of this section is to outline in detail the python code, which is broken down below as it pertains to the model and simulation creating a common understanding for all future users. The python code is very basic in nature utilizing the Networkx package in conjunction with Matplotlib to create the visual representation of the infection plots at the initial point and the end infection node status. The first step in the code development is installing the packages required within python to run the appropriate code. For the purpose of running the code the platform utilized in the following code was PyCharm, which is easily operational and can be accessed at the following location <https://www.jetbrains.com/pycharm>. The packages to be installed are Networkx Package, NumPy Package, Matplotlib Package and the CSV Package. The Networkx package provides the ability to graph the nodes and edges required for the network development. The NumPy package provides the mathematical capabilities for random number generation as well as statistical measures. The Matplotlib package provides the ability to graph in figures enabling the ability to visualize the graph in real time, and the CSV package is

utilized to read the CSV files with the network database. Additional packages are able to be added to the code if required to increase the performance of the simulation or model. The following excerpt displayed in figure is the code required for inputting the packages within PyCharm Coding Platform:

```
import networkx as nx
import numpy as np
import matplotlib.pyplot as plt
import csv
```

Figure 10 - Package Import Code

The section of code to be utilized are the inputs of the measurements utilized in the simulation which are agencies (comparing and displaying nodes infected on line plot), Infected (Probability of Infection to a given node), Susceptible (Susceptibility of the whole network), Initial Infection (the infection status of the first node created), and P_init_local (edge weight). Next the text `n_size = []` is utilized to establish a container for the nodes size variables utilized within the excel file for the levels of the agencies such as state, region, county and local. The code `G = nx. Graph ()` establishes the graph within the Networkx package and the `plt.figure(figsize=(20, 15))` determines the size of the figure when printed. Time is utilized for the start in the simulation timestep period. The `mFileName = desired fire` is required for drawing the data from the database file, and the counter starts at 0 when inputting all the nodes into the graph creating the linkage between the correct nodes. All of these coding requirements are displayed in the following excerpt.

```

Agencies = 1 # Number of patches
Infected = 0.0001 # Probability of Infection
Susceptible = 0.0001 # Probability of being Susceptible
Initial_Infection = 0.001 # Probability a patch will be
                        infected at the beginning
P_init_local = 0.005 # made this high to weight the
                        edge of the graph to start

n_size = []

G = nx. Graph ()
plt. figure (figsize= (20, 15))
Time = [0]
mFileName = "Florida_Law_Enforcement_Network.csv"
counter = 0

```

Figure 11 - Condition Setting Code

The next step to begin the development of the Network Architecture Model is to insert to create the beginning patch which establishes the node's status, and pos. The patch is created in a class enabling multiple patches (nodes) to be created, followed by the establishment of an object using def_init identifying the status and pos of a particular object or node. The code def_str(self) followed by return(str(self.status)) is used to return the status of a particular node where required to compare the probabilities against the infection status within the model. The following excerpt lays out the required code for the creation of the class.

```

class patch:
    def __init__ (self, status=0, pos=0):
        self. status = status
        self.pos = pos
    def __str__(self):
        return (str (self. status))

```

Figure 12 - Class Defining Code

Inserting data and creating the model is done so by using def LoadData (file) command pulling the information from the database. When developing the graph of the model the initial node is created using the global counter which starts at 0 and adds 1 for every next node that needs to be created within the model. The counter reads the CSV file for the next line within the file and respective node identification creating the connections based on the development of the database. The next line within the code for the data requirement is the initial status of the first node created where 1 is infected and 0 is normal. The determination of infection is calculated by comparing the random number generated between 0 and 1 against the Initial Infection value. P_Top is the top-level node using the code patch (Stat, counter) determining the infection status and count 0. The next line in the code establishes the first node within the graph where node = G.add_node(p_top) is the command to add node with no connections. The counter = counter + 1 establishes any following node addition must add 1 to the value in sequence creating the list. The last piece of code within the data load section establishes the size of the top-level node using n_size. append (10000) setting the size of the node to 10000 within the created graph. All of the required python code for the section is in the excerpt below.

```

def Load Data(file):
    global counter
    Stat = 1 if np. random. uniform () < Initial_Infection else 0
    Pos = (np. random. uniform () * 10 - 5, np. random. uniform () * 10 - 5)
    p_top = patch (Stat, counter)
    node = G.add_node (p_top, time='1pm', name="N-DEx")
    counter = counter + 1
    n_size. append (10000)

```

Figure 13 - Top Node Development and Counter Code

The code required for opening the CSV file within the model development for extracting the data is with open command as csvfile, followed by determining the reader requirement within python telling the package to read the file with the command reader = csv. DictReader(csvfile). The lines following those commands are required to create the required fields for the following code in the file.

```

with open (file, newline='') as csvfile:
    reader = csv. DictReader(csvfile)
    state = ""
    oldstate = ""
    state_counter = 0
    county_counter = 0
    region_counter = 0
    local_counter = 0
    region = ""
    oldregion = ""
    county = ""
    oldcounty = ""
    local = ""
    oldlocal = ""
    p_state = ""
    p_reg = ""
    p_count = ""

```

Figure 14 - Reading CSV File Code

Reading the CSV file requires code to determine how to read the file by row or column using the command: `for row in reader` followed by the line `state = row ['state']` providing the command to read from the state level down to the local level nodes within the CSV file. The command `if state != oldstate` is used to only populate one node at the state level which determines if the same state is listed it goes onto the next cell in the row to populate the next available node. The next line within the code for the data development is the status of the state node where 1 is infected and 0 is normal. The formula for this is `np.random.uniform() < float(row['state_infection'])` where the randomly generated number is compared to the infection probability located in the respective cell within the CSV file. The next aspect of the state development is establishing the state node with `p_state = patch (status and pos)` followed by the addition of the node at the state level. This is done using `G.add_node (p_state, name=state)` pulling the information from the CSV file identifying the state being read within the appropriate line. The next critical line of code attaches an edge from the top-level node to the state level node with the code `G.add_edge (p_top, p_state, weight=.25)` where the edge is created and the length of the edge is in relation to the weight. The next aspect of the code is to establish the size of the state level node by utilizing the code `n_size.append (int(row['state_node_size']))` which means the code is reading the respective cell under state node size as in integer creating the size of the state node. All of the required python code for the section is in the excerpt below outlining reading file code and state level code model development.

```

for row in reader:
    # row['county_infection'], row['local'], row['local_infection'])

    state = row['state']
    # state Level.
    # if it's a new state make a new top node.
    if state != oldstate:
        Stat = 1 if np.random.uniform() < float(row['state_infection']) else 0
        Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
        p_state = patch(Stat, Pos)
        G.add_node(p_state, name=state)
        G.add_edge(p_top, p_state, weight=.25)
        state_counter = counter
        counter = counter + 1
        oldstate = state
        n_size.append(int(row['state_node_size']))

```

Figure 15 - state Node Development Code

This section of code identifies the region level node with the following line `region = row['region']` providing the command to read from the region level down to the local level nodes within the CSV file. The command `if region != oldregion` is used to only populate one node at the region level which determines if the same region is listed or goes onto the next cell in the row to populate the next available node. The next line within the code for the data development is the status of the region node where 1 is infected and 0 is normal. The formula for this is `np.random.uniform() < float(row['region_infection'])` where the randomly generated number is compared to the infection probability located in the respective cell within the CSV file. The next aspect of the region development is establishing the region node with `p_region = patch(status and pos)` followed by the addition of the node at the region level. This is done using `G.add_node(p_region, name=region)` pulling the information from the CSV file identifying the region being read within the appropriate line. The next critical line of code attaches an edge from the state-level node to the region level node with the code `G.add_edge(p_state, p_region, weight=.25)` where the edge is created and the length of the edge is in relation to the weight. The next aspect

of the code is to establish the size of the region level node by utilizing the code `n_size.append(int(row['region_node_size']))` which means the code is reading the respective cell under region node size as in integer creating the size of the region node. All of the required python code for the region level node development section is in the excerpt below.

```

region = row['region']
if region != oldregion:
    # Stat = 1 if np.random.uniform() < region_infection else 0
    Stat = 1 if np.random.uniform() < float(row['region_infection']) else 0
    Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
    p_reg = patch(Stat, Pos)
    G.add_node(p_reg, name=region)
    G.add_edge(p_state, p_reg, weight=.25)
    region_counter = counter
    counter = counter + 1
    oldregion = region
    n_size.append(int(row['region_node_size']))

```

Figure 16 - region Node Development Code

This section of code identifies the county level node with the following line `county = row['county']` providing the command to read from the county level down to the local level nodes within the CSV file. The command `if county != oldcounty` is used to only populate one node at the county level which determines if the same county is listed or goes onto the next cell in the row to populate the next available node. The next line within the code for the data development is the status of the county node where 1 is infected and 0 is normal. The formula for this is `np.random.uniform() < float(row['county_infection'])` where the randomly generated number is compared to the infection probability located in the respective cell within the CSV file. The next aspect of the county development is establishing the county node with `p_county = patch(status and pos)` followed by the addition of the node at the county level. This is done using `G.add_node`

(p_county, name=county) pulling the information from the CSV file identifying the county being read within the appropriate line. The next critical line of code attaches an edge from the region-level node to the county level node with the code `G.add_edge (p_region, p_county, weight=.25)` where the edge is created and the length of the edge is in relation to the weight. The next aspect of the code is to establish the size of the county level node by utilizing the code `n_size.append (int(row['county_node_size']))` which means the code is reading the respective cell under county node size as in integer creating the size of the county node. All of the required python code for the county level node development section is in the excerpt below.

```

county = row['county']
    if county != oldcounty:
        # Stat = 1 if np.random.uniform() < county_infection else 0
        Stat = 1 if np.random.uniform() < float(row['county_infection']) else 0
        Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
        p_count = patch(Stat, Pos)
        G.add_node(p_count, name=county)
        G.add_edge(p_reg, p_count, weight=.25)
        county_counter = counter
        counter = counter + 1
        oldcounty = county
        n_size.append(int(row['county_node_size']))

```

Figure 17 - County Node Development Code

This section of code identifies the local level node with the following line `local = row ['local']` providing the command to read from the local level down to the local level nodes within the CSV file. The command `if local != oldlocal` is used to only populate one node at the local level which determines if the same local is listed or goes onto the next cell in the row to populate the next available node. The next line within the code for the data development is the status of the local node where 1 is infected and 0 is normal. The formula for this is `np.random.uniform()`

< float(row['local_infection'])) where the randomly generated number is compared to the infection probability located the respective cell within the CSV file. The next aspect of the local development is establishing the local node with p_local = patch (status and pos) followed by the addition of the node at the local level. This is done using G.add_node (p_local, name=local) pulling the information from the CSV file identifying the local being read within the appropriate line. The next critical line of code attaches an edge from the county-level node to the local level node with the code G.add_edge (p_region, p_local, weight=.25) where the edge is created and the length of the edge is in relation to the weight. The next aspect of the code is to establish the size of the local level node by utilizing the code n_size.append (int(row['local_node_size'])) which means the code is reading the respective cell under local node size as in integer creating the size of the local node. All of the required python code for the local level node development section is in the excerpt below.

```

local = row['local']
    if local != oldlocal:
        Stat = 1 if np.random.uniform() < float(row['local_infection']) else 0
        Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
        # local Level
        p_loc = patch(Stat, Pos)
        G.add_node(p_loc, name=local)
        G.add_edge(p_count, p_loc, weight=.25)
        local_counter = counter
        counter = counter + 1
        n_size.append(int(row['local_node_size']))

```

Figure 18 - local Node Development Code

After completing the code for the development of the model architecture the next step in the process is to establish the simulation steps pertaining to the infection status. To begin the

simulation code section, you must first define the operation with `def Simulate (Infection)`: this creates the platform for execution. The next line in sequence `for timestep in range (500)` defines the amount of times the simulation is cycled through. The next line `Status ()` is printing the status of each node after comparing the probabilities of each state, region, County, and local nodes from the formulas above. After the completion of the initial status the next lines are simply checking for the infection in the initial status which is then compared to the neighbors of each infected node. The next line `for n in G.nodes()`: simply identifies that the simulation is going to look at any node within the graph identified as G. Within this code the line `if status == 1` (not infected) and `np.random.uniform() < Infected`: then this identifies the node as infected utilizing the code `status = 1` (infected). After the initial infections have been identified the next step is for the simulation to look at all the nodes within the graph G, where if the node is infected using `if n.status == 0` then it lists out all the neighbors of the infected node. Once these nodes are identified the neighbor nodes are compared with a random number using `if np. random.uniform()` is compared against the Susceptible value set above, which is a fluctuating variable within the simulations between 0.0001, 0.001, and 0.01. Upon completion of the comparison of values and infection probabilities with the nodes the first timestep is complete and is then run another 499 times to account for the 500 timesteps. Each timestep executed in the simulation sequence is completed with the run function advancing the simulation one time by utilizing the code: `time. append (timestep + 1)` and the next line of code appends the infection simulation in order to plot each timestep of infected nodes over time. The required code conducting the simulation is in the following excerpt.

```

def Simulate(Infection):
    for timestep in range(500):
        Status()
        ## Check for infections
        for n in G.nodes():
            if n.status == 1 and np.random.uniform() < Infected:
                n.status = 1
        ## Check for Agencies that are infected
        for n in G.nodes():
            if n.status == 0:
                neighb = G[n] # That's it, a list of the neighbors
                for nei in neighb:
                    if nei.status == 0:
                        if np.random.uniform() < Susceptible:
                            nei.status = 1
                            break

        Time.append(timestep + 1)
        Infection.append(np.sum([n.status for n in G]) / float(Agencies))

```

Figure 19 - Simulation Code Development

Once all the above code is entered and checked for errors then both the model and simulation baseline code has been established and the next step in the process is to create and define the main operational code which executes the previous code created. The code to define the main operation is `def main():` followed by various commands to load the data as well as printing the CSV file information. The next critical piece of code listed within the main code section is identifying the pos and infection status with the code `occup = [n.status for n in G]`. After this section, the next area of code calculates the degree and centrality of the nodes within the graph. After this code is established the plots must be established to create the visual representations of the initial and post simulation infection statuses. The code used to create the initial graph is the `nx.draw()` command where you must input `G` as the graph, `node_size`, `node_color`, and labels to provide the information required for the graph. The command `plt.savefig()` is utilized to save the figure drawn at the time of creation and then the code `plt.show()` must be used after each desired graph creation to develop the graph itself. To better

develop a graph, reliable resources for matplotlib libraries are able to be viewed at https://matplotlib.org/2.1.1/api/_as_gen/matplotlib.pyplot.plot.html. To plot the graph with the infections of the nodes the first code to be used must be the Infection simulation code followed by the command `Simulate (Infection)` which inputs the simulation result into the graph being created. The next steps are followed as outlined above for the initial graph to populate the graph with the infected nodes identified. After both the initial plot and the infected plot have been established the next step is to create the line plot chart to provide a clear understanding of the infected nodes over the course of the timesteps. The critical code for establishing the line plot is `plt.plot(Time, Infection)`, where the data used for developing the graph is Time on X axis and the simulation data on the Y axis laying out the infected nodes over the 1000 timesteps. The commands `plt.xlabel`, `plt.ylabel`, `plt.grid`, and `plt.savefig` can be used to develop an understanding of the graph. The command `plt.show()` must be used to create the graph. The following excerpt clearly outlines the code requirements for execution the simulation file.

```

def main():
    print("Loading Data from file.... ", mFileName)
    LoadData(mFileName)
    pos = {}
    for n in G.nodes():
        pos[n] = n.pos
    occup = [n.status for n in G]

    plt.title('Initial Infection Status', fontsize=30)
    nx.draw(G, center=1581, node_size=(n_size), node_color=occup, with_labels=False, cmap=plt.cm.plasma,
            vmin=0, vmax=1)

    plt.show()

    Infection = [np.sum([n.status for n in G]) / float(Agencies)]
    Simulate(Infection)

    plt.figure(figsize=(20, 15))
    plt.title('Post Simulation Infection Status', fontsize=30)
    nx.draw(G, center=1581, node_size=(n_size), node_color=[n.status for n in G], with_labels=False,
            cmap=plt.cm.plasma, vmin=0, vmax=1)

    plt.show()

if __name__ == "__main__":
    main()

```

Figure 20 - Main Execution Code

The complete set of code is able to be referenced in Appendix C compiling all of the above outlined python code for model development as well as simulation execution which allows for direct implementation into any system.

3.6 Simulation Execution and Data Collection Methodology

Conducting the Florida, Pennsylvania, Federal level, and Federal level Web-Based Law Enforcement Network Simulations was done so in a strategic manner to allow for data production and analysis. This section outlines procedures utilized to conduct the simulations, layout the procedures for data organization, and the methodology for achieving the averages for comparison. The simulation is conducted by each network architecture setup (Florida, Pennsylvania, Federal level, Federal level Web-Based) with 4000 simulation runs per the county

and local infection probabilities which are outlined in the data development section above. After one simulation run the data output for initial node infection and total node infection are recorded. This is executed 4000 times per data set identified above as well in the Simulation Infection Probability Parameter Table Figure. Upon completion of all simulation runs the data must be organized for easy comparison across the data sets to provide functional analysis. To provide a functional base for comparison the data is organized by Parameters and Results to achieve common understanding of the data structure. The Parameters data section is organized by the following categories Agencies, Timesteps, Probability of Infection, Probability of being Susceptible, Probability of Initial Infection, Initial Edge Weight, state Infection Probability, region Infection Probability, County Infection Probability, and the local Infection Probability. Following the Parameters section is the results section which contains the Number of Nodes in the Graph, Number of Edges in the Graph, Average Degree, Network Density, Initial Node Infection, Rate of Infection, and the Nodes Infected. Both of these data areas are paired with a line plot for each simulation that has been ran which provides detail for each simulation run. These can be found in Appendix A though Appendix D. An example of the informational data Parameters and results associated with each of the simulations can be viewed in the following table.

Table 4 - Parameters and Results Output Example

Parameters	Agencies = 1
	TimeSteps = 1000
	Probability of Infection = 0.0001
	Probability of being Susceptible = 0.0001
	Probability of Initial Infection = 0.001
	Initial Edge Weight = 0.005
	State Infection Probability = 0.001
	Region Infection Probability = 0.01
	County Infection Probability = (RANDBETWEEN(0,15)/100)
	Local Infection Probability = (RANDBETWEEN(0,25)/100)
Results	Number of Nodes = 277
	Number of Edges = 276
	Average Degree = 1.9928
	Network Density = 0.007220216606498195
	Initial Node Infection = 26.0
	Rate of Infection = 0.06 (6.0%)
	Nodes Infected = 68.0

The Rate of Infection seen in the figure above is calculated first by finding the Node Infection (NI) which is the Total Node Infection value at timestep 500 minus the Initial Node Infection value delivering the Node Infection (NI) value.

$$NI = \text{Node Infection @ TS 500} - \text{Initial Node Infection} \quad (1)$$

The next step is to calculate the Node Infection Rate (NIR) which is the Node Infection (NI) value divided by 500 identifying the timestep analysis delivering the NIR. Both of these equations can be seen in the below figures clearly identify the represented values utilized for the calculation. The first equation defined is

$$NIR = \frac{NI}{500} \quad (2)$$

Upon completion of each data set simulation consisting of the 4000 iterations the average for the required measurable values are able to be calculated. The values that are used to compare

the simulation data sets are the Average Initial Node Infection, the Average Rate of Infection, and the Average Nodes Infected. The Average Initial Node Infection (AINI) is calculated by taking the value NI then divided by 4000 (number of simulations run in the iteration) to calculate the Average Initial Node Infection. The equation is displayed in the figure below:

$$AINI = \frac{NI}{4000} \quad (3)$$

After the calculation of the NI and NIR values the Average Infection Rate (AIR) can be calculated by finding the SUM of the Node Infection Rate (NIR) where the 4000 simulation infection rates of one iteration are added to produce a total value. This value is then divided by 4000 to deliver the AIR. The equation is displayed in the following figure below:

$$AIR = \frac{SUM(NIR)}{4000} \quad (4)$$

The Average Total Nodes Infected (ANI) is calculated by finding the SUM of the 4000 simulations Node Infected values producing a total value number divided by 4000 representing the simulation within the data set iteration. The equation is displayed in the following figure below:

$$ANI = \frac{SUM(Total\ Nodes\ Infected)}{4000} \quad (5)$$

Calculation of all the above values enables the platform to be established for analysis of the average data sets within each infection probability set to be compared in relation to the Susceptibility values against other network architectures. In conjunction with node infection rates and average initial infections the ability to compare against network density is critical. The network density is a representation of the connections within any given network architecture. The denser the architecture of the network the easier it is for information to flow as compared to a network with a sparse structure which constricts the flow of information based on the limiting connections. The utilization of Network Density in the Law Enforcement Networks comparison allows for the demonstration of information flow based on the given architecture within the selected states and federal government. Network density is calculated utilizing two key equations which provide the foundation for the numerical value demonstrating the density of a given network. The first step is to determine the number of connections within a given network by taking number of possible connections (N) and multiplying it with N-1 and then dividing by 2, which provides the number of Potential Connections (PC) displayed below in equation 8. The next step is to take the number of Actual Connections (AC) and then divide that by the Potential Connection (PC) determining the network density of the selected architecture demonstrated in equation 9 below.

$$PC = N * \frac{N-1}{2} \quad (6)$$

$$Network\ Density = \frac{AC}{PC} \quad (7)$$

3.6.1 Statistical Data Development and Comparison

At the completion of development and data output of the above equations, the ability to derive the statistical outputs is driven by the functions of utilizing excel. The purpose of using excel for deriving a statistical data set for each simulation run iteration based on the initial node infection, node infection at timestep 500, and infection is to provide the ability to cross examine the data sets for each susceptibility range as well as probability sets. The method for data development is utilized with the function (data analysis) within excel under the data tab accessing the descriptive statistics method. This method under the data analysis tab provides the ability to derive the summary statistics within the descriptive setting as well as the confidence level for the mean set at .95 (95%). The summary statistics within the descriptive statistics function provides the data for the following: mean, standard error, median, mode, standard deviation, sample variance, kurtosis, skewness, range, minimum, maximum, $\mu + 1\sigma$, $\mu - 1\sigma$, $\mu + 2\sigma$, $\mu - 2\sigma$, and confidence level. The mean defines the average of the data set, the standard error is the measure of the statistical accuracy of an estimate, the median is the midpoint of a frequency distribution and the mode is the most reoccurring numerical value. The standard deviation is the distance from the mean, the kurtosis is the sharpness of the peak of a frequency-distribution curve and the skewness is the amount of deviation from the normal distribution. All of the above-outlined statistics are accompanied by a histogram of the 4000 iterations and the results plotted with a normal distribution identifying the standard deviations as well as the mean on the visible chart. The purpose of this measure is to ensure the data is identifiable in the

simulation runs conducted allowing for cross analysis. The ability to compare the deviations as well as the averages among the susceptibilities and probabilities ensures the platform for infection measurements. All of the data sets are conducted in this manner to identify the statistical measures. Figure 18 below outlines a sample of the histogram model utilized as well as the data set table accompanying the data set for analysis.

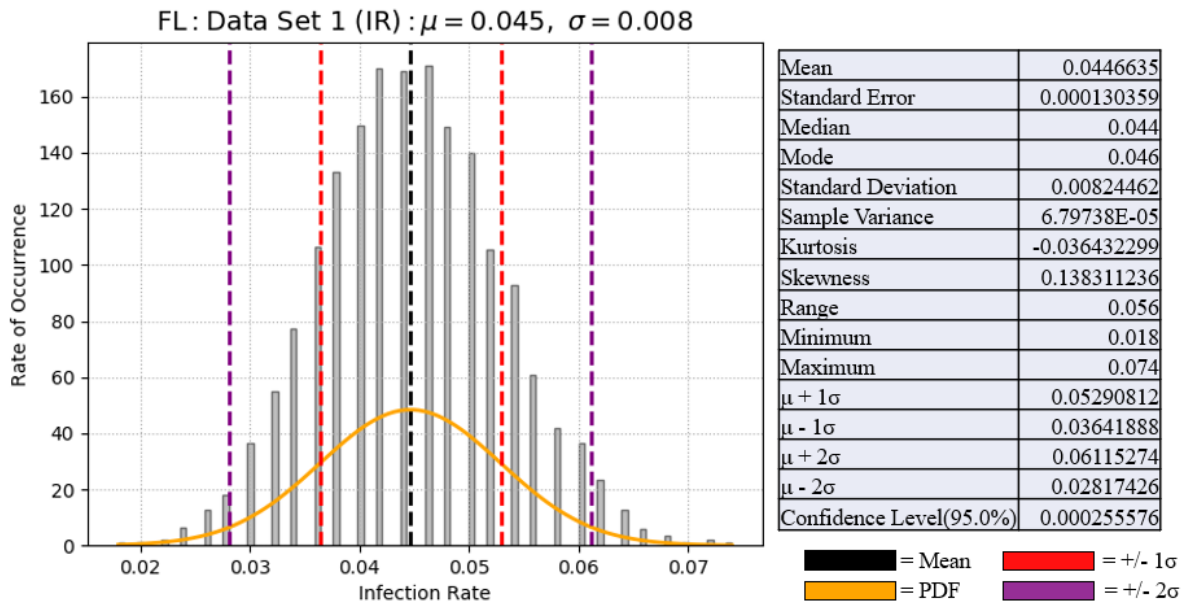


Figure 21 - Histogram Data Set and Statistics Table

Given these methods of analyzing the statistical outcomes is based on the confidence level of 95 percent across the board. The ability to identify the average infection rate across the data sets creates the ability to understand the infection across the various network architectures. Understanding and outlining the architectures enables the simulation to produce the result-oriented data that enables correlation analysis with the structure, nodes, probabilities, and susceptibility of network infections.

CHAPTER 4: RESULTS

This section presents the data from the conducted simulations based on each network architecture in correlation with the designed data sets. Tables of statistical outputs will follow pertaining to the executed simulations.

4.1 Network Architecture Statistical Comparison

There are four network architectures utilized during the execution of the simulation to provide the foundation for determining the infection rate (spread) based on the susceptibility of the network, topology of the network and the infection probabilities at the various law enforcement agency nodes. The four architectures utilized as outlined in chapter 3 are the Florida Law Enforcement Network, Pennsylvania Law Enforcement Network, Federal Law Enforcement Network, and the Federal Web-Based Law Enforcement Network. Understanding the fundamentals for degrees and density provides the foundation for comparison among the various network structures. The average degree as referred within this thesis examines the number of connections other nodes (Hagberg, Schult, & Swart, 2018). The network density refers to the portion of potential connections in a network that are actual connections (Hagberg, Schult, & Swart, 2018), which is explained in chapter 3 in equations 6 and 7. The **Florida Law Enforcement Network** contains 277 nodes, 276 edges, an average degree of 1.9928, and a network density of 0.007220216606498195. The **Pennsylvania Law Enforcement Network** contains 285 nodes, 284 edges, an average degree of 1.9930, and a network density of

0.007017543859649123. The **Federal Law Enforcement Network** contains 561 nodes, 560 edges, an average degree of 1.9928, and a network density of 0.0035650623885918. The **Federal Web-Based Law Enforcement Network** contains 405 nodes, 404 edges, an average degree of 1.9951, and a network density of 0.0049382716049382715. Breaking down each network architecture provides the fundamental understanding of the compositions as well as densities enabling comprehension of the information flow capabilities as it relates to the spread of infection. The Law Enforcement Model of Network Susceptibility (LEMONS) 10 total simulation iterations per network architecture are based on the data sets outlined in chapter 3 table 6 which are broken down into the statistical data result explanations in the following sections. The breakdown of statistical data encompasses the Standard Deviation (SD), the Standard Error (SE), the Kurtosis, and the Skewness. The kurtosis regarding the Law Enforcement Model of Network Susceptibility (LEMONS) will demonstrate the sharpness of the frequency-distribution curve (Groeneveld & Meeden, Dec., 1984). The skewness is a measure of the asymmetry of the probability distribution of a real-values random variable, which can be valued as positive, negative or undefined (Groeneveld & Meeden, Dec., 1984). The overall findings and statistical comparisons are able to be viewed in the following table below.

Table 5 - Network Infection Rate Comparison

	Florida	Pennsylvania	Federal	Federal Web-Based
Nodes	277	285	561	405
Edges	276	284	560	404
Density	0.00722	0.00701	0.00356	0.00493
Average Degree	1.9928	1.9930	1.9928	1.9951
Infection Rate Comparison				
Data Set 1	0.045	0.082	0.08	0.008
Data Set 2	0.19	0.231	0.345	0.008
Data Set 3	0.239	0.243	0.435	0.008
Data Set 4	0.039	0.072	0.105	0.008
Data Set 5	0.167	0.205	0.402	0.008
Data Set 6	0.214	0.218	0.433	0.008
Data Set 7	0.029	0.054	0.061	0.008
Data Set 8	0.133	0.164	0.270	0.008
Data Set 9	0.17	0.177	0.354	0.008
Data Set 10	0.046	0.084	0.109	0.008
	Florida	Pennsylvania	Federal	Federal Web-Based

The complete statistical breakdown overview in correlation with the graphed results for each iteration are available in Appendix D (**Florida Law Enforcement Network**), Appendix E (**Pennsylvania Law Enforcement Network**), Appendix F (**Federal Law Enforcement Network**), and Appendix G (**Federal Web-Based Law Enforcement Network**).

4.2 Florida Law Enforcement Network Simulation Results

This section provides the results of the Law Enforcement Model of Network Susceptibility (LEMONS) for the Florida Network Architecture. The results are outlined from simulation one through the tenth simulation.

The **first simulation** conducted is based on **Data Set # 1**, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 5 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 130 – 131.

Table 6 - Florida Network Simulation Results Data Set 1

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.812	2.393	0.0378	6	5	0.295	0.499	18	0	18
Infection @ TS 500	28.143	4.565	0.0721	28	29	-0.006	0.175	31	14	45
Infection Rate	0.045	0.008	0.0001	0.044	0.046	-0.036	0.138	0.056	0.018	0.074

The **second simulation** conducted is based on **Data Set # 2**, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 6 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 131 – 132.

Table 7 - Florida Network Simulation Results Data Set 2

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.604	2.359	0.0373	5	5	0.109	0.371	15	0	15
Infection @ TS 500	100.592	5.594	0.0884	101	101	-0.066	0.071	38	83	121
Infection Rate	0.19	0.011	0.0001	0.19	0.19	-0.071	0.095	0.074	0.156	0.23

The **third simulation** conducted is based on **Data Set # 3**, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 7 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 133 – 134.

Table 8 - Florida Network Simulation Results Data Set 3

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.475	2.291	0.0362	5	5	0.111	0.408	16	0	16
Infection @ TS 500	124.734	6.949	0.1098	125	122	-0.179	-0.013	45	101	146
Infection Rate	0.239	0.014	0.0002	0.238	0.238	-0.099	0.025	0.098	0.188	0.286

The **fourth simulation** conducted is based on **Data Set # 4**, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 8 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 134 – 135.

Table 9 - Florida Network Simulation Results Data Set 4

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	30.632	5.121	0.0809	30	30	0.083	0.171	39	13	52
Infection @ TS 500	50.150	5.889	0.0931	50	49	-0.075	0.005	40	32	72
Infection Rate	0.039	0.008	0.0001	0.038	0.036	-0.042	0.122	0.052	0.014	0.066

The **fifth simulation** conducted is based on **Data Set # 5**, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 9 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 136 – 137.

Table 10 - Florida Network Simulation Results Data Set 5

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	30.043	5.078	0.0803	30	28	0.000	0.111	40	12	52
Infection @ TS 500	113.509	6.163	0.0974	113	113	0.018	0.049	44	91	135
Infection Rate	0.167	0.012	0.0001	0.168	0.17	-0.037	0.004	0.084	0.126	0.21

The **sixth simulation** conducted is based on **Data Set # 6**, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 10 below; the associated graphical outputs

for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 137 – 138.

Table 11 - Florida Network Simulation Results Data Set 6

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	30.279	5.067	0.0801	30	31	-0.063	0.134	36	15	51
Infection @ TS 500	137.501	7.148	0.1131	138	138	-0.195	-0.027	46	114	160
Infection Rate	0.214	0.014	0.0002	0.214	0.216	-0.114	0.027	0.094	0.166	0.26

The **seventh simulation** conducted is based on **Data Set # 7**, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 11 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 139 – 140.

Table 12 - Florida Network Simulation Results Data Set 7

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	85.231	7.26	0.114	85	86	-0.133	-0.041	46	61	107
Infection @ TS 500	99.963	7.355	0.116	100	98	-0.054	-0.072	50	76	126
Infection Rate	0.029	0.007	0.0001	0.03	0.028	0.039	0.231	0.048	0.008	0.056

The **eighth simulation** conducted is based on **Data Set # 8**, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The

results found from the simulation are outlined in table 12 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 140 – 141.

Table 13 - Florida Network Simulation Results Data Set 8

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	81.487	6.805	0.107	81	80	0.029	0.058	49	58	107
Infection @ TS 500	147.771	6.121	0.096	148	149	-0.066	-0.024	46	126	172
Infection Rate	0.133	0.011	0.0001	0.132	0.13	-0.055	0.022	0.082	0.098	0.18

The **ninth simulation** conducted is based on **Data Set # 9**, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 13 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 142 – 143.

Table 14 - Florida Network Simulation Results Data Set 9

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	85.498	6.598	0.104	85	84	-0.101	0.101	45	65	110
Infection @ TS 500	170.554	5.822	0.092	171	172	0.152	-0.093	45	146	191
Infection Rate	0.17	0.012	0.0001	0.17	0.166	0.134	-0.004	0.082	0.13	0.212

The **tenth simulation** conducted is based on **Data Set # 10**, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.0001, region Nodes = to 0.0001, County Nodes = to 0.0001

and the local Nodes = to 0.0001(only 1 node infected at .25 infection probability). The results found from the simulation are outlined in table 14 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 143 – 144.

Table 15 - Florida Network Simulation Results Data Set 10

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	0.283	0.47	0.0074	0	0	0.248	1.226	3	0	3
Infection @ TS 500	23.433	4.217	0.0666	23	23	0.004	0.136	31	10	41
Infection Rate	0.046	0.008	0.0001	0.046	0.044	-0.014	0.126	0.06	0.02	0.08

4.3 Pennsylvania Law Enforcement Network Simulation Results

This section provides the results of the Law Enforcement Model of Network Susceptibility (LEMONS) for the Pennsylvania Network Architecture. The results are outlined from the eleventh simulation through the twentieth simulation.

The **eleventh simulation** conducted is based on Data Set # 1, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 15 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 146 – 147.

Table 16 - Pennsylvania Network Simulation Results Data Set 1

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.732	2.369	0.0374	6	5	-0.102	0.29	14	0	14
Infection @ TS 500	46.817	5.323	0.0841	47	46	0.053	0.084	36	30	66
Infection Rate	0.082	0.01	0.0001	0.082	0.082	0.127	0.064	0.074	0.048	0.122

The **twelfth simulation** conducted is based on Data Set # 2, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 16 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 147 – 148.

Table 17 - Pennsylvania Network Simulation Results Data Set 2

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.679	2.346	0.0371	6	6	0.029	0.355	15	0	15
Infection @ TS 500	120.626	6.92	0.1094	121	116	-0.067	0.072	50	95	145
Infection Rate	0.231	0.014	0.0002	0.23	0.23	-0.054	0.101	0.096	0.188	0.284

The **thirteenth simulation** conducted is based on Data Set # 3, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 17 below; the associated graphical outputs

for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 149 – 150.

Table 18 - Pennsylvania Network Simulation Results Data Set 3

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	5.707	2.346	0.0371	6	5	0.291	0.434	17	0	17
Infection @ TS 500	127.38	7.488	0.1184	127	127	-0.048	-0.015	52	104	156
Infection Rate	0.243	0.015	0.0002	0.244	0.244	0.042	-0.011	0.11	0.19	0.3

The **fourteenth simulation** conducted is based on Data Set # 4, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 18 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 150 – 151.

Table 19 - Pennsylvania Network Simulation Results Data Set 4

	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	28.686	4.98	0.0787	29	28	0.067	0.084	38	11	49
Infection @ TS 500	64.496	6.116	0.0967	64	65	0.138	0.061	45	42	87
Infection Rate	0.072	0.011	0.0001	0.072	0.072	-0.035	0.081	0.064	0.042	0.106

The **fifteenth simulation** conducted is based on Data Set # 5, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The

results found from the simulation are outlined in table 19 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 150 – 151.

Table 20 - Pennsylvania Network Simulation Results Data Set 5

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	28.783	4.971	0.0781	29	28	-0.082	0.175	35	12	47
Infection @ TS 500	131.072	6.966	0.1101	131	131	-0.113	0.055	48	108	156
Infection Rate	0.0205	0.014	0.0002	0.204	0.208	-0.037	0.054	0.116	0.152	0.268

The **sixteenth simulation** conducted is based on Data Set # 6, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 20 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 153 – 154.

Table 21 - Pennsylvania Network Simulation Results Data Set 6

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	28.613	4.987	0.0788	29	29	-0.091	0.121	34	13	47
Infection @ TS 500	137.607	7.55	0.1193	138	140	0.019	-0.002	53	110	163
Infection Rate	0.218	0.015	0.0002	0.218	0.216	0.074	0.054	0.108	0.164	0.272

The **seventeenth simulation** conducted is based on Data Set # 7, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes =

to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100).

The results found from the simulation are outlined in table 21 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 155 – 156.

Table 22 - Pennsylvania Network Simulation Results Data Set 7

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	90.24	6.741	0.1066	90	89	0.026	0.007	51	67	118
Infection @ TS 500	117.131	6.858	0.1084	117	115	-0.119	-0.047	45	95	140
Infection Rate	0.054	0.009	0.0001	0.054	0.052	-0.109	0.139	0.056	0.026	0.082

The **eighteenth simulation** conducted is based on Data Set # 8, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 22 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 156 – 157.

Table 23 - Pennsylvania Network Simulation Results Data Set 8

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	90.473	6.925	0.1095	90	90	-0.074	0.015	54	61	115
Infection @ TS 500	172.591	6.117	0.096	173	173	-0.051	-0.027	45	148	193
Infection Rate	0.164	0.012	0.0001	0.164	0.166	-0.001	0.041	0.08	0.124	0.204

The **nineteenth simulation** conducted is based on Data Set # 9, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 23 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 158 – 159.

Table 24 - Pennsylvania Network Simulation Results Data Set 9

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	90.423	6.933	0.1096	91	92	-0.042	-0.068	51	65	116
Infection @ TS 500	179.102	6.193	0.0979	179	178	-0.011	-0.044	41	158	199
Infection Rate	0.177	0.012	0.0001	0.178	0.18	-0.01	0.128	0.082	0.14	0.222

The **twentieth simulation** conducted is based on Data Set # 10, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.0001, region Nodes = to 0.0001, County Nodes = to 0.0001 and the local Nodes = to 0.0001(only 1 node infected at .25 infection probability). The results found from the simulation are outlined in table 24 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 159 – 160.

Table 25 - Pennsylvania Network Simulation Results Data Set 10

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	0.245	0.442	0.0069	0	0	0.568	1.384	3	0	3
Infection @ TS 500	42.298	5.027	0.0794	42	42	0.061	0.017	38	24	62
Infection Rate	0.084	0.01	0.0001	0.084	0.084	0.047	0.017	0.076	0.048	0.124

4.4 Federal Law Enforcement Network Simulation Results

This section provides the results of the Law Enforcement Model of Network Susceptibility (LEMONS) for the Federal Network Architecture. The results are outlined from the twenty-first simulation through the thirtieth simulation.

The **twenty-first simulation** conducted is based on Data Set # 1, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 25 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 162 – 163.

Table 26 - Federal Network Simulation Results Data Set 1

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	61.858	7.171	0.1133	62	63	0.012	0.111	53	37	90
Infection @ TS 500	101.829	8.295	0.1311	102	104	-0.113	0.001	55	74	129
Infection Rate	0.08	0.011	0.0001	0.08	0.08	0.065	0.028	0.084	0.042	0.126

The **twenty-second simulation** conducted is based on Data Set # 2, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 26 below; the associated graphical outputs

for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 163 – 164.

Table 27 - Federal Network Simulation Results Data Set 2

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	61.798	7.151	0.1131	62	62	-0.067	0.113	51	36	87
Infection @ TS 500	234.081	8.678	0.1372	234	231	-0.047	0.051	64	203	267
Infection Rate	0.345	0.016	0.0003	0.344	0.346	0.003	0.093	0.116	0.292	0.408

The **twenty-third simulation** conducted is based on Data Set # 3, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 27 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 165 – 166.

Table 28 - Federal Network Simulation Results Data Set 3

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	62.110	7.161	0.1132	62	63	0.168	0.030	59	28	87
Infection @ TS 500	279.406	10.23	0.1617	280	282	-0.206	0.012	69	245	314
Infection Rate	0.435	0.020	0.0003	0.434	0.43	-0.095	0.015	0.142	0.356	0.498

The **twenty-fourth simulation** conducted is based on Data Set # 4, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100).

The results found from the simulation are outlined in table 28 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 166 – 167.

Table 29 - Federal Network Simulation Results Data Set 4

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	61.354	7.143	0.113	61	60	-0.021	0.103	47	39	86
Infection @ TS 500	101.308	8.303	0.131	101	103	0.037	-0.005	66	64	130
Infection Rate	0.105	1.611	0.025	0.08	0.076	3999.62	63.241	101.96	0.04	102

The **twenty-fifth simulation** conducted is based on Data Set # 5, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 29 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 168 – 169.

Table 30 - Federal Network Simulation Results Data Set 5

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	61.287	7.304	0.115	61	60	0.012	0.107	54	35	89
Infection @ TS 500	232.791	9.101	0.143	233	232	32.072	-1.676	203	60	263
Infection Rate	0.402	3.694	0.058	0.344	0.346	3999.83	63.243	233.7	0.278	234

The **twenty-sixth simulation** conducted is based on Data Set # 6, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to

(RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 30 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 169 – 170.

Table 31 - Federal Network Simulation Results Data Set 6

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	61.34	7.189	0.113	61	61	-0.038	0.072	50	38	88
Infection @ TS 500	277.633	10.34	0.163	278	277	-0.062	-0.033	73	239	312
Infection Rate	0.433	0.021	0.0003	0.432	0.440	-0.011	0.016	0.138	0.364	0.502

The **twenty-seventh simulation** conducted is based on Data Set # 7, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 31 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 171 – 172.

Table 32 - Federal Network Simulation Results Data Set 7

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	170.211	9.472	0.149	170	169	-0.021	0.014	71	135	206
Infection @ TS 500	200.73	9.885	0.156	201	201	0.043	0.036	81	162	243
Infection Rate	0.061	0.010	0.0001	0.06	0.062	0.086	0.105	0.072	0.028	0.1

The **twenty-eighth simulation** conducted is based on Data Set # 8, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 32 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 172 – 173.

Table 33 - Federal Network Simulation Results Data Set 8

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	170.004	9.419	0.148	170	171	0.057	-0.013	72	133	205
Infection @ TS 500	305.073	8.687	0.137	305	307	0.051	-0.059	65	270	335
Infection Rate	0.270	0.016	0.0002	0.270	0.268	-0.068	-0.024	0.11	0.214	0.324

The **twenty-ninth simulation** conducted is based on Data Set # 9, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 33 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 174 – 175.

Table 34 - Federal Network Simulation Results Data Set 9

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	169.93	9.523	0.151	170	169	0.051	0.004	73	137	210
Infection @ TS 500	346.779	8.58	0.135	347	346	-0.121	-0.113	54	317	371
Infection Rate	0.354	0.017	0.0002	0.354	0.35	-0.167	0.037	0.116	0.296	0.412

The **thirtieth simulation** conducted is based on Data Set # 10, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.0001, region Nodes = to 0.0001, County Nodes = to 0.0001 and the local Nodes = to 0.0001(only 1 node infected at .25 infection probability). The results found from the simulation are outlined in table 34 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 175 – 176.

Table 35 - Federal Network Simulation Results Data Set 10

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	0.308	0.497	0.0078	0	0	1.125	1.309	4	0	4
Infection @ TS 500	47.528	5.987	0.0946	47	48	0.935	0.012	67	0	67
Infection Rate	0.109	0.947	0.0149	0.094	0.096	3998.74	63.23	59.95	0.054	60

4.5 Federal Web-Based Law Enforcement Network Simulation Results

This section provides the results of the Law Enforcement Model of Network Susceptibility (LEMONS) for the Federal Web-Based Network Architecture. The results are outlined from the thirty-first simulation through the fortieth simulation.

The **thirty-first simulation** conducted is based on Data Set # 1, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 35 below; the associated graphical outputs

for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 178 – 179.

Table 36 - Federal Web-Based Network Simulation Results Data Set 1

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	10.213	3.156	0.0498	10	9	-0.011	0.289	20	2	22
Infection @ TS 500	14.284	3.776	0.0597	14	13	0.380	0.456	28	4	32
Infection Rate	0.008	0.004	0.0001	0.008	0.006	3.231	1.492	0.032	0.002	0.034

The **thirty -second simulation** conducted is based on Data Set # 2, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The results found from the simulation are outlined in table 36 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 179 – 180.

Table 37 - Federal Web-Based Network Simulation Results Data Set 2

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	10.401	3.136	0.0495	10	10	-0.189	0.259	19	2	21
Infection @ TS 500	14.398	3.646	0.0576	14	14	0.099	0.344	25	4	29
Infection Rate	0.008	0.004	0.0001	0.008	0.006	3.666	1.474	0.034	0.002	0.036

The **thirty -third simulation** conducted is based on Data Set # 3, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,2)/100) and the local Nodes = to (RANDBETWEEN (0,5)/100). The

results found from the simulation are outlined in table 37 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 181 – 182.

Table 38 - Federal Web-Based Network Simulation Results Data Set 3

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	10.395	3.164	0.0500	10	10	0.072	0.251	23	1	24
Infection @ TS 500	14.377	3.252	0.0514	14	14	0.007	0.221	24	4	28
Infection Rate	0.008	0.002	0.0001	0.008	0.008	1.763	0.418	0.014	0.002	0.016

The **thirty -fourth simulation** conducted is based on Data Set # 4, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 38 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 182 – 183.

Table 39 - Federal Web-Based Network Simulation Results Data Set 4

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	51.304	6.541	0.1034	51	49	0.131	0.180	48	30	78
Infection @ TS 500	55.242	6.853	0.1083	55	53	0.186	0.198	56	34	90
Infection Rate	0.008	0.004	0.0001	0.006	0.006	2.097	1.298	0.028	0.002	0.03

The **thirty -fifth simulation** conducted is based on Data Set # 5, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to

(RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 39 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 184 – 185.

Table 40 - Federal Web-Based Network Simulation Results Data Set 5

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	51.189	6.573	0.1039	51	48	0.009	0.155	45	29	74
Infection @ TS 500	55.207	6.825	0.1079	55	55	0.013	0.145	47	34	81
Infection Rate	0.008	0.004	0.0001	0.008	0.006	2.347	1.326	0.026	0.002	0.028

The **thirty -sixth simulation** conducted is based on Data Set # 6, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,15)/100) and the local Nodes = to (RANDBETWEEN (0,25)/100). The results found from the simulation are outlined in table 40 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 185 – 186.

Table 41 - Federal Web-Based Network Simulation Results Data Set 6

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	51.276	6.537	0.1033	51	50	0.133	0.056	45	29	74
Infection @ TS 500	55.246	6.589	0.1041	55	55	0.108	0.041	47	32	79
Infection Rate	0.008	0.002	0.0001	0.008	0.008	1.434	0.429	0.014	0.002	0.016

The **thirty -seventh simulation** conducted is based on Data Set # 7, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 41 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 187 – 188.

Table 42 - Federal Web-Based Network Simulation Results Data Set 7

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	147.755	8.745	0.1382	148	147	-0.005	-0.043	60	119	179
Infection @ TS 500	151.780	8.902	0.1407	152	150	0.061	-0.021	63	121	184
Infection Rate	0.008	0.004	0.0001	0.008	0.004	3.263	1.473	0.036	0.002	0.038

The **thirty -eighth simulation** conducted is based on Data Set # 8, utilizing the susceptibility equal to 0.001, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 42 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 188 – 189.

Table 43 - Federal Web-Based Network Simulation Results Data Set 8

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	147.531	8.443	0.1335	147	148	0.036	0.012	63	116	179
Infection @ TS 500	151.549	8.661	0.1369	152	152	0.101	0.049	69	118	187
Infection Rate	0.008	0.004	0.0001	0.008	0.006	2.925	1.428	0.03	0.002	0.032

The **thirty -ninth simulation** conducted is based on Data Set # 9, utilizing the susceptibility equal to 0.01, state Nodes = to 0.001, region Nodes = to 0.01, County Nodes = to (RANDBETWEEN (0,25)/100) and the local Nodes = to (RANDBETWEEN (0,75)/100). The results found from the simulation are outlined in table 43 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 190 – 191.

Table 44 - Federal Web-Based Network Simulation Results Data Set 9

Measured Statistical Ouputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	147.572	8.522	0.1347	148	148	-0.037	-0.023	63	116	179
Infection @ TS 500	151.558	8.591	0.1358	152	153	-0.026	-0.013	62	120	182
Infection Rate	0.008	0.002	0.0001	0.008	0.008	2.192	0.835	0.022	0.002	0.024

The **fortieth simulation** conducted is based on Data Set # 10, utilizing the susceptibility equal to 0.0001, state Nodes = to 0.0001, region Nodes = to 0.0001, County Nodes = to 0.0001 and the local Nodes = to 0.0001(only 1 node infected at .25 infection probability). The results found from the simulation are outlined in table 44 below; the associated graphical outputs for the Initial Infection, Infection at Timestep 500, and Infection Rate can be reference in Appendix D on pages 191 – 192.

Table 45 - Federal Web-Based Network Simulation Results Data Set 10

Measured Statistical Outputs								Descriptive Statistics		
	Mean	SD	SE	Median	Mode	Kurtosis	Skewness	Range	Min	Max
Initial Infection	0.294	0.481	0.0076	0	0	0.228	1.219	2	0	2
Infection @ TS 500	4.342	2.067	0.0327	4	3	2.099	1.315	12	2	14
Infection Rate	0.008	0.004	0.0001	0.008	0.006	2.291	1.394	0.026	0.002	0.028

Although all the above tables clearly identify and layout the statistical results per data set by network structure the ability to cross reference infection rates by network is needed. The infection rate comparison provides the ability to understand the infection rate in relation to the node count as well as the network design increasing the foundation. Table 45 below outlines each network outlined above providing the infection rate organized by data for easy comparison.

Table 46 - Network Infection Rate Comparison

	Florida	Pennsylvania	Federal	Federal Web-Based
Nodes	277	285	561	405
Edges	276	284	560	404
Density	0.00722	0.00701	0.00356	0.00493
Average Degree	1.9928	1.9930	1.9928	1.9951
Infection Rate Comparison				
Data Set 1	0.045	0.082	0.08	0.008
Data Set 2	0.19	0.231	0.345	0.008
Data Set 3	0.239	0.243	0.435	0.008
Data Set 4	0.039	0.072	0.105	0.008
Data Set 5	0.167	0.205	0.402	0.008
Data Set 6	0.214	0.218	0.433	0.008
Data Set 7	0.029	0.054	0.061	0.008
Data Set 8	0.133	0.164	0.270	0.008
Data Set 9	0.17	0.177	0.354	0.008
Data Set 10	0.046	0.084	0.109	0.008
	Florida	Pennsylvania	Federal	Federal Web-Based

Based on the ten data sets when viewing Table 45 analyzing the infection rate increase by topology and infection probability is able to be outlined as follows. **Data set one** shows an infection rate increase by 0.037 (3.7%) from Florida to Pennsylvania, an infection rate increase by 0.035 (3.5%) from Florida to the Federal Network, an infection rate decrease by -0.002 (-0.2%) from Pennsylvania to the Federal Network, and a -0.072 (-7.2%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set two** shows an infection rate increase by 0.041 (4.1%) from Florida to Pennsylvania, an infection rate increase by 0.155 (15.5%) from Florida to the Federal Network, an infection rate increase by 0.114 (11.4 %) from Pennsylvania to the Federal Network, and a -0.337 (-33.7%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set three** shows an infection rate increase by 0.004 (0.4%) from Florida to Pennsylvania, an infection rate increase by 0.196 (19.6%) from Florida to the Federal Network, an infection rate increase by 0.192 (19.2 %) from Pennsylvania to the Federal Network, and a -0.427 (-42.7%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set four** shows an infection rate increase by 0.033 (3.3%) from Florida to Pennsylvania, an infection rate increase by 0.066 (6.6%) from Florida to the Federal Network, an infection rate increase by 0.033 (3.3 %) from Pennsylvania to the Federal Network, and a -0.097 (-9.7%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set five** shows an infection rate increase by 0.038 (3.8%) from Florida to Pennsylvania, an infection rate increase by 0.235 (23.5%) from Florida to the Federal Network, an infection rate increase by 0.197 (19.7 %) from Pennsylvania to the Federal Network, and a -0.394 (-39.4%) infection rate decrease from the

Federal Network to the Federal Web-Based Topology. **Data set six** shows an infection rate increase by 0.004 (0.4%) from Florida to Pennsylvania, an infection rate increase by 0.219 (21.9%) from Florida to the Federal Network, an infection rate increase by 0.215 (21.5 %) from Pennsylvania to the Federal Network, and a -0.425 (-42.5%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set seven** shows an infection rate increase by 0.025 (2.5%) from Florida to Pennsylvania, an infection rate increase by 0.032 (3.2%) from Florida to the Federal Network, an infection rate increase by 0.007 (0.7 %) from Pennsylvania to the Federal Network, and a -0.053 (-5.3%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set eight** shows an infection rate increase by 0.031 (3.1%) from Florida to Pennsylvania, an infection rate increase by 0.137 (13.7%) from Florida to the Federal Network, an infection rate increase by 0.106 (10.6 %) from Pennsylvania to the Federal Network, and a -0.262 (-26.2%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set nine** shows an infection rate increase by 0.007 (0.7%) from Florida to Pennsylvania, an infection rate increase by 0.184 (18.4%) from Florida to the Federal Network, an infection rate increase by 0.177 (17.7 %) from Pennsylvania to the Federal Network, and a -0.346 (-34.6%) infection rate decrease from the Federal Network to the Federal Web-Based Topology. **Data set ten** shows an infection rate increase by 0.038 (3.8%) from Florida to Pennsylvania, an infection rate increase by 0.063 (6.3%) from Florida to the Federal Network, an infection rate increase by 0.025 (2.5 %) from Pennsylvania to the Federal Network, and a -0.101 (-10.1%) infection rate decrease from the Federal Network to the Federal Web-Based Topology.

Based on the above outlined results the increase in infection rates from Florida to Pennsylvania is consistent among all ten data sets based in the 8-node increase from the Florida topology to the Pennsylvania topology. The average increase in infection rate between Florida and Pennsylvania utilizing the 0.0001 the average susceptibility parameter is 0.0316 (3.16%), for the 0.001 the average susceptibility parameter is 0.0366 (3.66%) and the 0.01 average susceptibility parameter is 0.005 (0.5%). The increase in infection rates from Florida topology to the Federal topology is consistent among all ten data sets based in the 284-node increase from the Florida topology to the Federal topology. The average increase in infection rate between Florida and the Federal level utilizing the 0.0001 susceptibility parameter is 0.044 (4.4%), the 0.001 susceptibility parameter is 0.175 (17.5%) and the 0.01 susceptibility parameter is 0.199 (19.9%). The increase in infection rates from Pennsylvania topology to the Federal topology is consistent among all ten data sets based in the 276-node increase from the Pennsylvania topology to the Federal topology. The average increase in infection rate between Pennsylvania and the Federal level utilizing the 0.0001 susceptibility parameter is 0.012 (1.2%), the 0.001 susceptibility parameter is 0.139 (13.9%) and the 0.01 susceptibility parameter is 0.194 (19.4%). The decrease in infection rates from the Federal level topology to the Federal Web-Based topology is consistent among all ten data sets based in the 156-node decrease from the Federal level topology to the Federal Web-Based topology. The average increase in infection rate between Federal level topology to the Federal Web-Based topology utilizing the 0.0001 susceptibility parameter is -0.074 (-7.4%), the 0.001 susceptibility parameter is -0.331 (-33.1%) and the 0.01 susceptibility parameter is -0.399 (-39.9%).

CHAPTER 5: SUMMARY OF FINDINGS

This section provides a summary of the results, contributions to the research, issues, and limitations encountered during the research, and the potential for future research as well as the implementation of the Law Enforcement Model of Network Susceptibility (LEMONS).

5.1 Results and Contributions

Based on the cybersecurity risks and issues on the rise in correlation to the Law Enforcement community networks, the research of infection probabilities at the various law enforcement levels is truly critical and possess the potential to directly affect the security of the Law Enforcement Network structures at all levels and agencies. This work and research were focused on the shifting network topologies in relation to the infection probabilities identifying the infection rates in correlation to the virus injected within a specific network. The simulated virus was utilized in the form of the susceptibility of the network in conjunction with the random number generation compared to the infection probabilities determining the initial status of each and every node within the simulation.

5.1.1 Infection Rate Comparison

The results comprised in chapter 4 can be analyzed based on the node counts and infection rates based on the data sets by network structure. There are several observations that can be made according to the data outcome, where the first major observation is based on the number of nodes within a given network architecture will increase the infection rate based on the connections available to increase the spread of a virus. This is clearly identified when viewing table 45 in chapter 4, which provides the comparison for each topology utilized identifying the web-based topology as the lowest risk for infection spread. The second observation is the level of infection probabilities at each of the node levels dramatically alters the infection rate based on the network topology in relation to the infection probability levels. The nodes located at the local level pose a greater risk for infection as the data sets are cycled through produce higher levels of infection rates based on the `RANDBETWEEN (0,75)100`, which identifies the highest level of infection probability within the simulation. The nodes at the county level pose the second greatest threat for infection spread based on the infection probability within the data sets.

Based on the above statistical analysis provided in chapter 4, one can conclude by utilizing a topology where network communications pass through from the local level to the county level to the region and to the state creates a more vulnerable platform for infection to spread based on the paths for information to travel increasing risk. The conclusion proven with statistical data regardless of the vulnerability of the nodes at various level the web-based approach is the most secure network topology to be utilized creating less intrusion risks when utilizing a singular tunnel for information sharing, while simultaneously monitoring the security

of the tunnel. Even with vulnerable nodes the mitigation of spread is provided based on the security of the main hub node ensuring increased network security.

5.1.2 Research Question I (RQ1) Results

The answer to RQ I is the infection probability at the Federal, state, region, County and local Nodes directly impact the spread of the infection. The infection is spread at higher rates throughout the topology if the infection probabilities are higher among the nodes creating more risk for not only initial infection but also infection throughout the simulation. It should be noted that for the simulation the probabilities selected were essential in the comprehension of spread impacts among law enforcement agency vulnerabilities (nodes) as the lower infection probabilities did, in fact, mitigate the infection spread at a much higher rate.

5.1.3 Research Question II (RQII) Results

The answer to RQ II is shifting the topology alone, whether increasing/decreasing the node count or utilizing the web-based topology does in fact directly affect the infection rate. Adding nodes within a network structure increases the infection spread due to the increased network density (connections) among the nodes. However, the exception, by increasing the node count and eliminating the path of travel for information travel utilizing the web-based topology mitigates the infection spread dramatically by approximately -0.101 (10.1%) from the Federal Network utilizing the susceptibility 0.0001 and 0.0001 infection probability for all nodes. The susceptibility alteration does, in fact, increase the infection as the susceptibility increases from

0.0001 to 0.001 to 0.01 which varies based on the topology utilized. In relation to each other the topology and susceptibility, increases the infection rate based on node counts as well as overall susceptibility. The overall impact of the topology and susceptibility provides the foundation for analysis amongst the selected network in relation to the overall security providing the ability to input real world network security measures into the simulation.

5.1.4 Research Question III (RQIII) Results

The answer to RQ III is by modifying network topology does in fact increase network security based on the statistical outcomes outlined in the above section 5.1.1 Infection Rate Comparison. The modification of the topologies by eliminating edges increases the network security mitigating the infection rate. This is compared to having multiple information tunnels for collaboration versus forcing all information to travel through one tunnel of flow which is closely managed for cyber spear phishing threats.

5.1.5 Research Question IV(RQIV) Results

The answer to RQ IV is the information flow among the various network topologies is determined based on the edges within the specific network creating the path of travel for information. In the Florida, Pennsylvania, and Federal Networks the configurations are similar where the information travels from the lowest level node through three levels of nodes to reach the N-DEX node, and the Federal Web-Based the information must on travel through one node to reach the N-DEX node. The impact of this is information flow is constricted when in a complex

network versus non-complex (web-based) as well as taking into account the infection probabilities per node creating a very complex model to understand the effects of viruses within the network.

Understanding the results and comparison of the topologies presented provides the foundation for real-world data to be entered into the Law Enforcement Model of Network Susceptibility (LEMONS) platform the following two sections 5.2 and 5.3 outline the proposed methods to establish the real-world infection probabilities for law enforcement agencies.

5.2 Network Vulnerability Measurement

In the current Cyber operational domain, the ability to acquire and utilize real time network data from law enforcement agencies are considered to be a risk based on the sensitivity of the data as well as the classification of the model to execute the analysis. Therefore, the infection probabilities are all assumed (fabricated using Microsoft Office Excel) to achieve realistic comparable results. The development of real data based on the vulnerability of a specified law enforcement agency network would be calculated based on the research from *Metrics of Security* (Cheng, Deng, Li, Deloach, & Singhai, 2012) identifying the Vulnerable Host Percentage (VHP). The VHP metric can represent the overall security level of a desired network. The number of vulnerable hosts within a desired law enforcement network is able to be derived by conducting a scan of a network utilizing a vulnerability scanning tool, which identifies the number of active ports. The equation for deriving the VHP is represented below in Equation 1 modified from *Metrics of Security* (Cheng, Deng, Li, Deloach, & Singhai, 2012).

Equation 1 represents the identified network, V is the sum of vulnerable hosts, and H is the sum of all hosts within the desired law enforcement network (Cheng, Deng, Li, Deloach, & Singhai, 2012).

$$VHP(G) = \frac{100 \times \frac{V}{H}}{2} \quad (8)$$

The vulnerable host percentage equation has been simplified to the lowest level to allow for simple and quick real data collection as well as input into the database. The ability to input real, accessible data would provide a platform for all agencies to utilize in order to assess their cyber readiness against other agencies. After identifying the vulnerabilities associated with a specified network the next step is to identify the severity of each vulnerability identified based on the Common Vulnerability Scoring System (CVSS) which is identified in Table 2 below reprinted from *Metrics of Security* (Cheng, Deng, Li, Deloach, & Singhai, 2012).

Table 47 - Severity levels of Vulnerabilities

CVSS Score	Severity Level	Guidance
7.0 through 10.0	High Severity	Must be corrected with the highest priority
4.0 through 6.9	Medium Severity	Must be corrected with high priority
0.0 through 3.9	Low Severity	Encouraged, but not required, to correct these vulnerabilities

Note. Reprinted from “Metrics of Security”, by Cheng, Y., Deng, J., Li, J., Deloach, S., & Singhai, A., 2012, National Institute of Standards and Technology

The past research conducted in relation to the vulnerability of the network as it pertains to the infection probabilities provides the foundation for enhancing the ability to input reliable, measurable, and effective data in to the simulation. Understanding the research conducted provides the ability to effectively calculate the infection in conjunction with the human factor research outlined in the section below. It should be noted this has been introduced as a stepping model with the intent to develop a probability model for real time infection data.

5.3 Human Factor Vulnerabilities Analysis

The Human Factor contribution to the model development has been researched in depth using surveys and questionnaires to assess the vulnerabilities within various organizations. This research methodology was conducted by Sten Mases outlined in the report Evaluation Method for Human Aspects of Information Security which outlines critical aspects of the vulnerabilities relating to the human in the loop risks (Mäses, 2015). The first aspect of this research is relevant to the Law Enforcement Model with the understanding of the vulnerabilities associated with the human factors concept among the informational platforms of the law enforcement community creating large risks for sensitive data. The evaluation method outlined in this paper address the knowledge-attitude-behavior (KAB) model which iterates the role of knowledge as it pertains to the logical relation to the intentionally related behaviors, where the KAB model suggests the behavior of an individual changes gradually (Mäses, 2015). It is important to highlight this study states clearly the belief the relationship between knowledge, attitude and behavior is influenced by many individuals, intervention and organizational factors (Mäses, 2015). The KAB model

was used to create a prototype for assessing information security awareness and as it pertains to the Law Enforcement Model Design would contribute to the calculation of the infection probability for the law enforcement organizations (Mäses, 2015).

The use of the KAB model provided the foundation for the ability to develop and evaluation of the human factor relating to information security by examining defined characteristics (Mäses, 2015). The examination questions identified within this report measured the security conscious behavior pertaining to information security were open-ended addressing information sharing, sensitive information handling and password management (Mäses, 2015). Although the method to change security awareness is the intent of this research paper the Law Enforcement Model Design is only interested in the risk-oriented behaviors relating to the probability values. In an attempt to discover the knowledge, attitudes and behaviors relating to the issue of information security within a desired network 35 questions were developed regarding the Golden rules (Mäses, 2015) as outlined in Chapter 1. The 35 questions were derived relating to 21 sub-areas outlined in the paper Evaluation Method for Human Aspects of Information Security which provided the feedback for the information security vulnerability or as it relates to the Law Enforcement Model Design infection probabilities values respective to the node levels. The questions utilized for the relative exam are located within the document which are necessary to be implemented within any law enforcement organization to determine the levels of knowledge and experience within the cyber operational domain.

Although the above research was conducted to implement a survey and experiment with ability to derive results-based data for the Law Enforcement Model into a scalable table is essential to achieve a probability for analysis. The methodology for the development of the

criteria table, as well as the grading table for the purpose of finding a desired probability has been developed which is outlined in “A Human Error Probability Estimate” by C.S. do Nascimento and R.N. de Mesquita (Nascimento & Mesquita, 2009). This research provides the fundamental understanding and association with the education levels as well as experience on the job creating a measurable scale. The development of variable groups within the criteria provides the ability to assess knowledge level in correlation with years of experience (Nascimento & Mesquita, 2009). The grading table was introduced to weight the data composition creating the ability to formulate the total for both Network Education as well as Experience as it relates to the Law Enforcement Network Model (Nascimento & Mesquita, 2009). The Network Security Vulnerability Assessment Criteria was modified from the original in the research paper to address the probability of infection statuses within the Law Enforcement Model. The table can be viewed below.

Table 48- Network Security Vulnerability Assessment Criteria (modified from (Nascimento & Mesquita, 2009)

NETWORK SECURITY EDUCATION	POINTS (P1)	EXPERIENCE (years)	POINTS (P2)
Masters	1	28 to 35	1
Graduate	2	21 to 27	2
Bachelors	3	14 to 20	3
High School	4	7 to 13	4
None	5	1 to 6	5

The modification of the criteria table provided the foundation for the development of the grading table Network Security Vulnerability Grading Table which was derived from the above

identified research paper regarding assessment. The Network Security Vulnerability Grading Table can be viewed below.

Table 49- Network Security Vulnerability Grading Table (modified from (Nascimento & Mesquita, 2009))

OFFICER (Li)	NETWORK SECURITY EDUCATION	POINTS (P1)	EXPERIENCE (years)	POINTS (P2)	PEi (P1 + P2)
L1	Master	1	30	1	2
L2	Graduate	2	32	1	3
L3	Graduate	2	27	2	4
L4	High School	4	18	3	7
L5	Bachelors	3	10	4	7
L6	High School	4	12	4	8
L7	None	5	10	4	9
TOTAL = $\sum PE_i$					40

The foundation provided by C.S. do Nascimento and R.N. de Mesquita in the development of the tables above leads directly to the equation formulation which was modified from the original variation to meet the output requirements of the infection-based probabilities within the Law Enforcement Model Design. The equation represented for the Human Factor Network Vulnerability Value (HFNVV) is calculated by taking the PE_i value (P1 + P2) which is then divided by the $\sum PE_i$ (total value of all possible PE_i combinations). The outcome is then multiplied by 2.777775 to account for 50 percent of the infection probability values based on the assumption the Human in the Loop factor is significant. The explained equation above is outlined below:

$$HFNVV = \frac{PE_i}{\sum PE_i} * 2.777775 \quad (9)$$

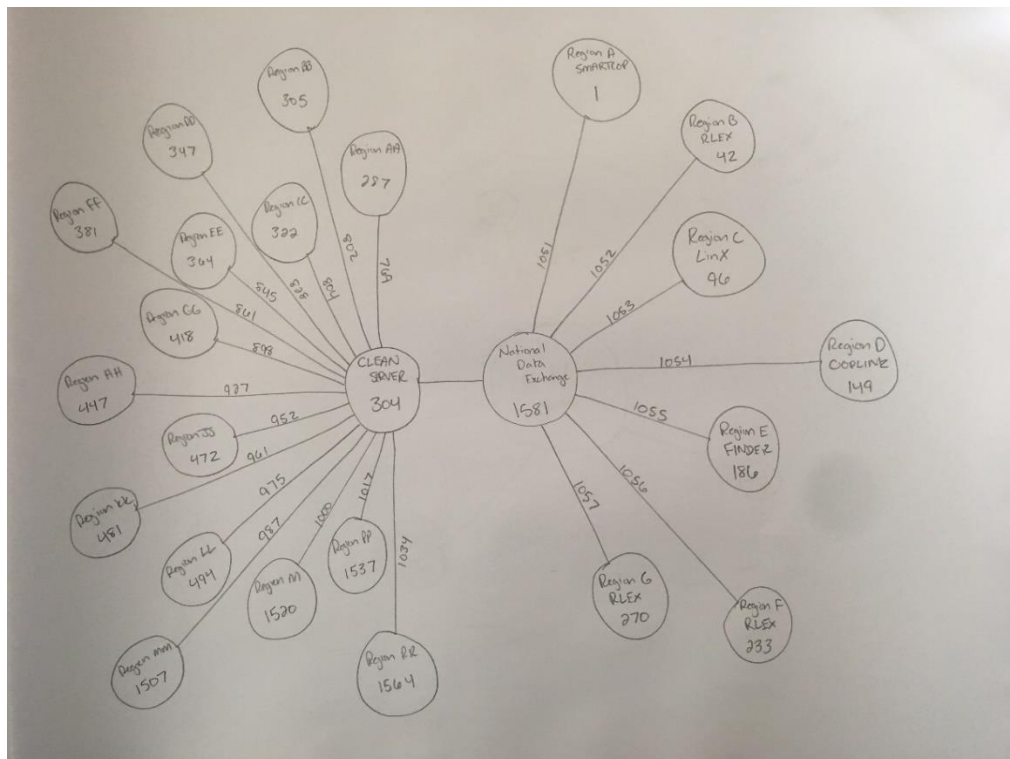
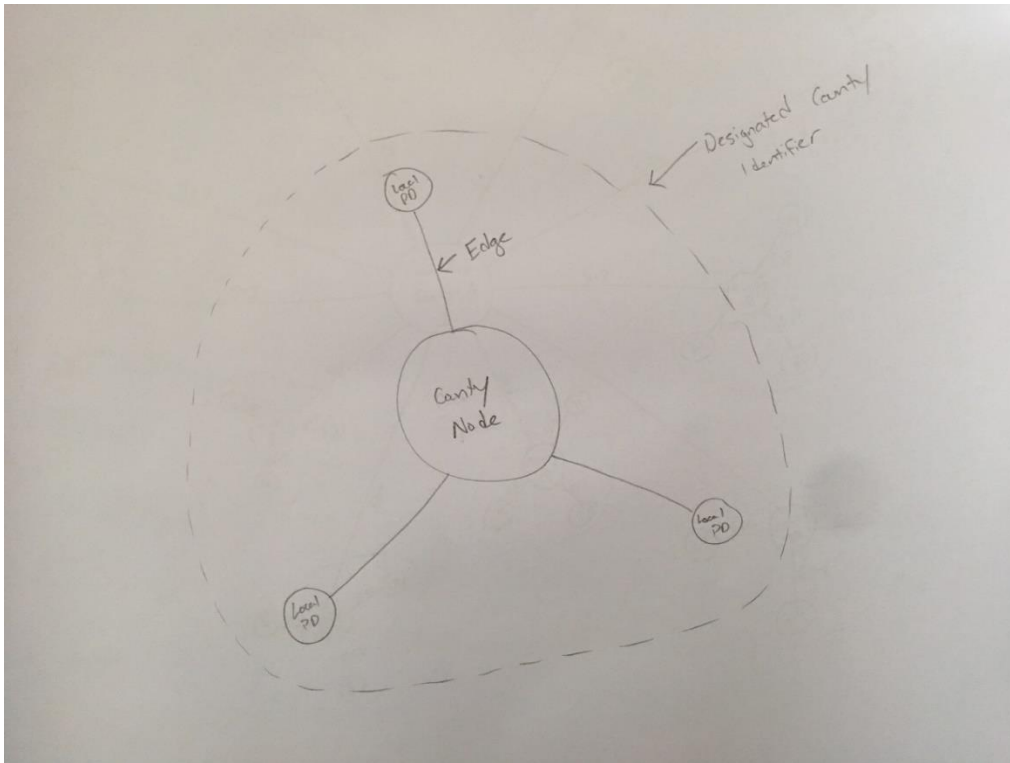
5.4 Future Work and Limitations

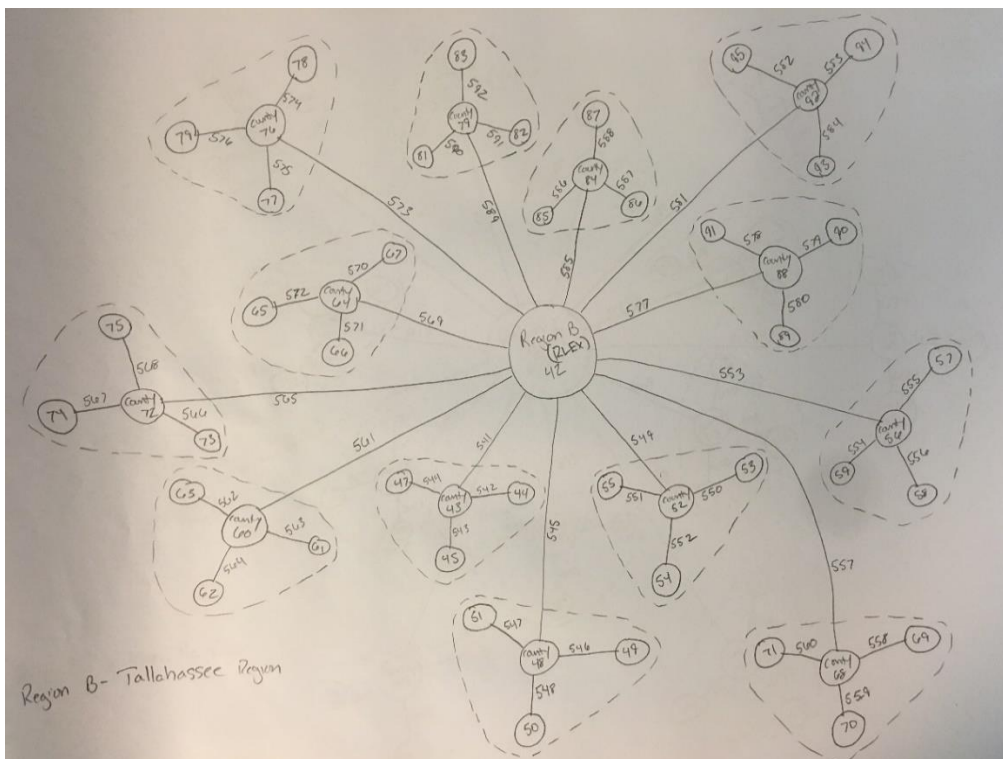
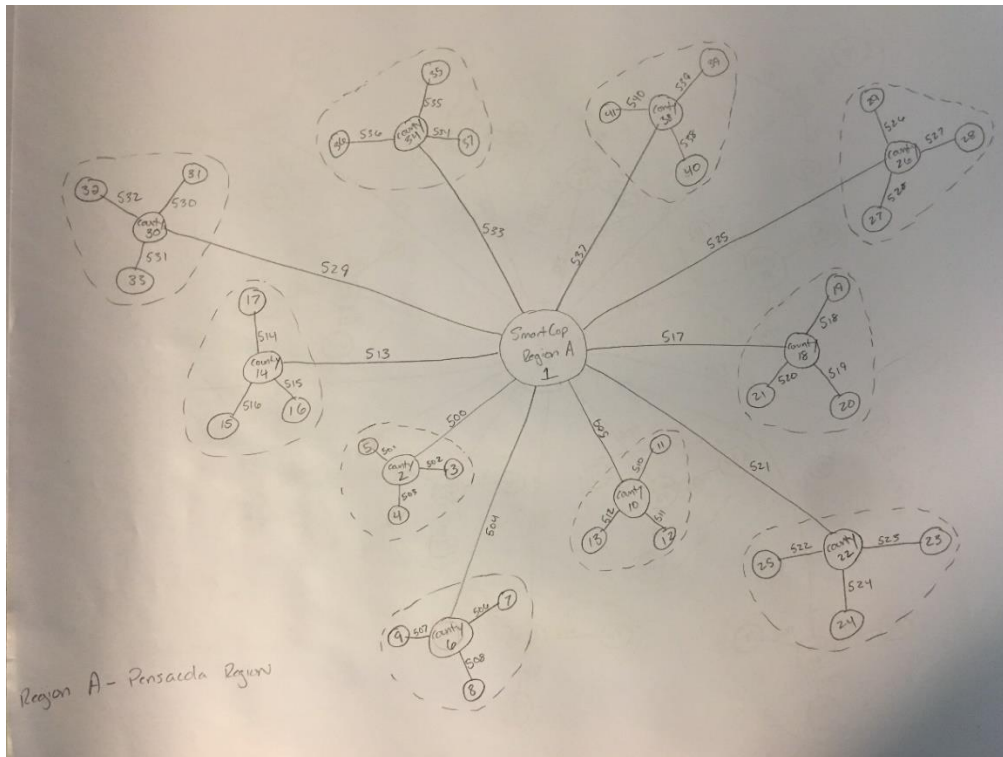
The Law Enforcement Model of Network Susceptibility (LEMONS) product has been developed in such a manner that future work is able to be conducted in conjunction with already established databases as well as code structures. The work that could be implemented in the future is firstly the ability to establish the code for recovery of an infected network node creating the ability to alter the infection rate over the timestep periods within the simulation. In conjunction with the recovery aspect of the code the database can be altered to increase the state count as well as the local node increasing the complexity of the model to more accurately replicate the live network. The critical aspect of the future work is that it is not limited but rather enhanced based on the technological changes to come within the cyber community. The ability to implement unforeseen desired advantages for the simulation aspect to increase the understanding of the law enforcement network vulnerabilities.

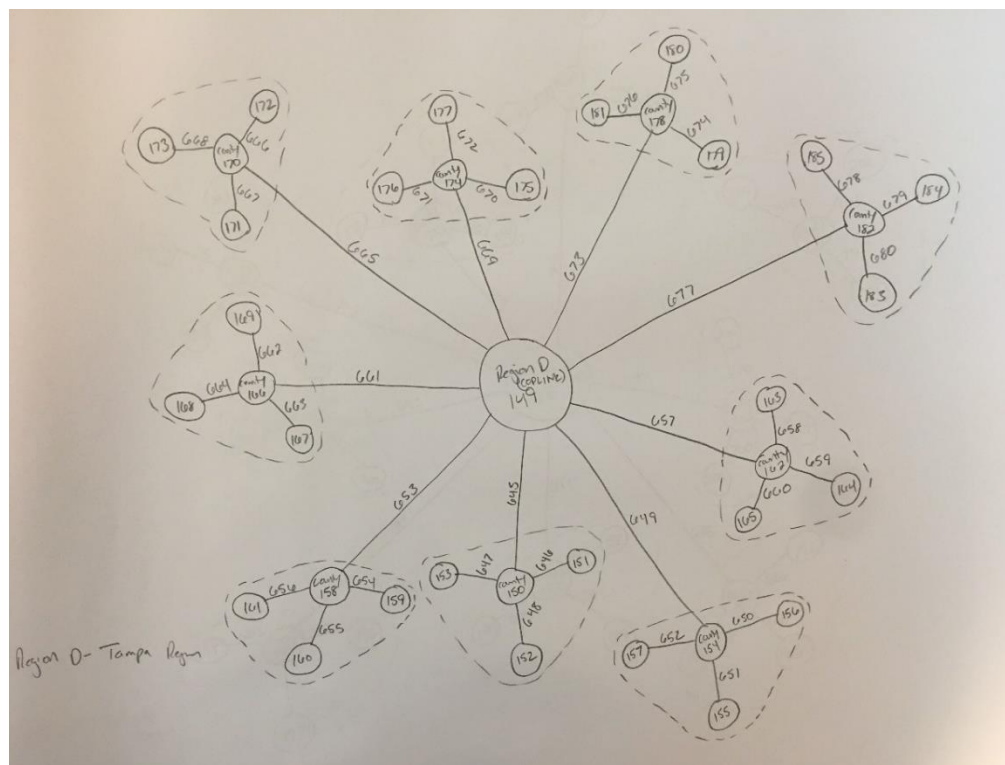
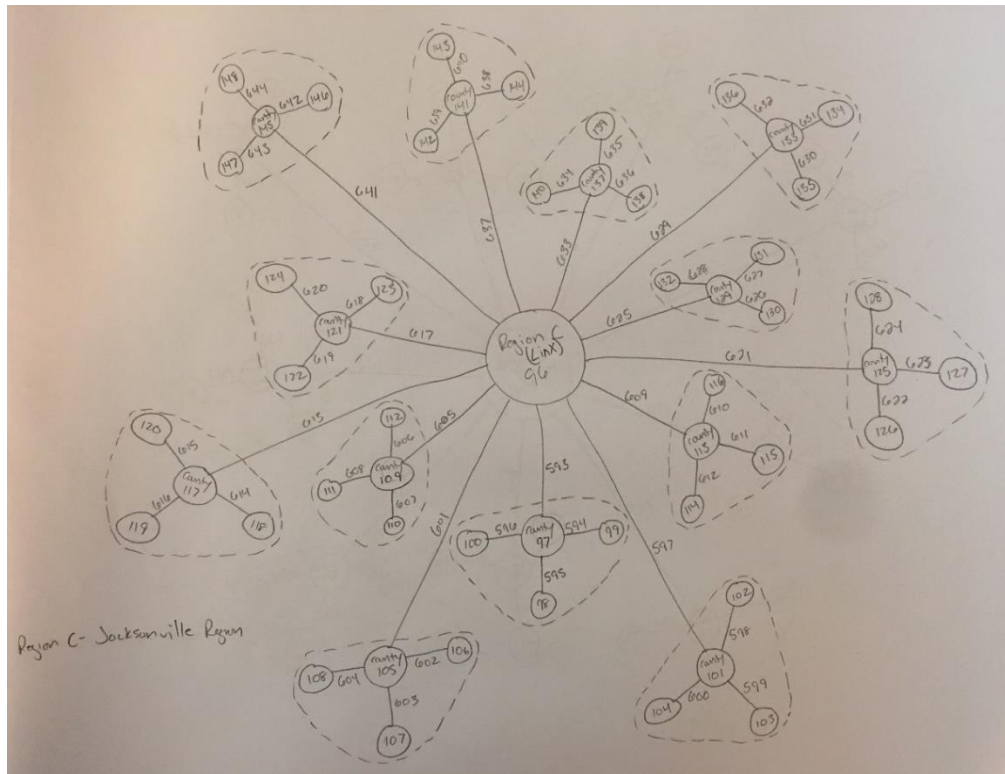
The limitations that occurred within the timeframe of conducting the research, design and implementation of this thesis project were a wide range of aspects. The first limitation encountered during the development of the Law Enforcement Model of Network Susceptibility (LEMONS) was limited coding knowledge and ability creating various road blocks as well as time-consuming weeks to complete the design. The second limitation encountered during the research and development was the lack of established similar model designs creating the need to develop the model from the ground up to provide the foundation for the CSV file usage allowing

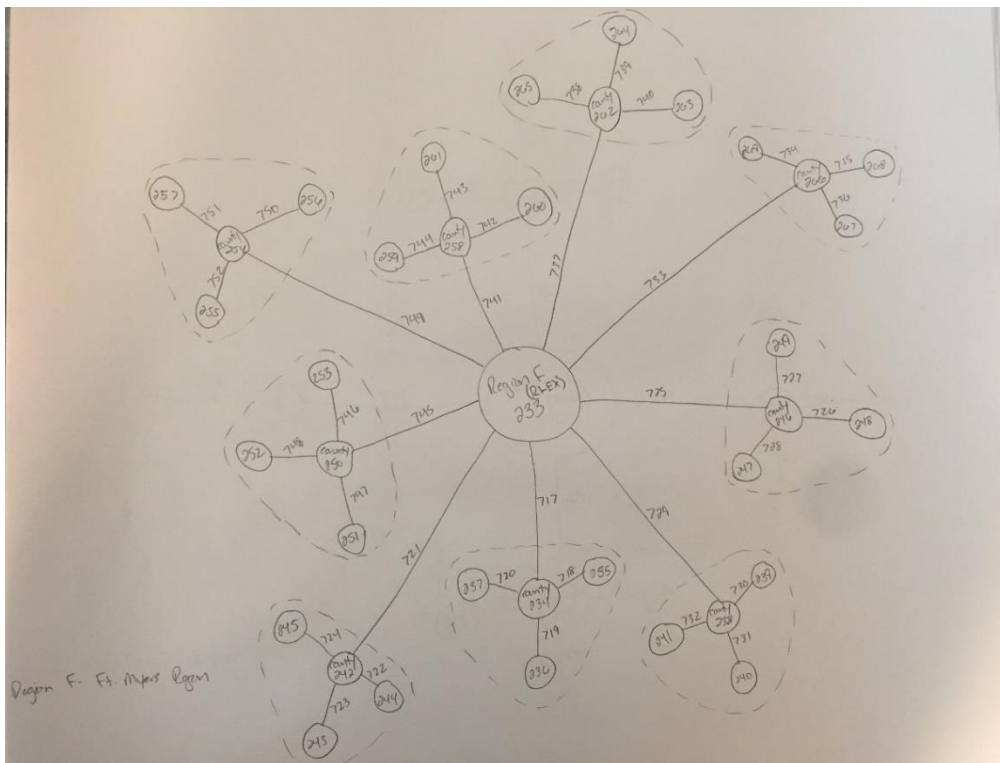
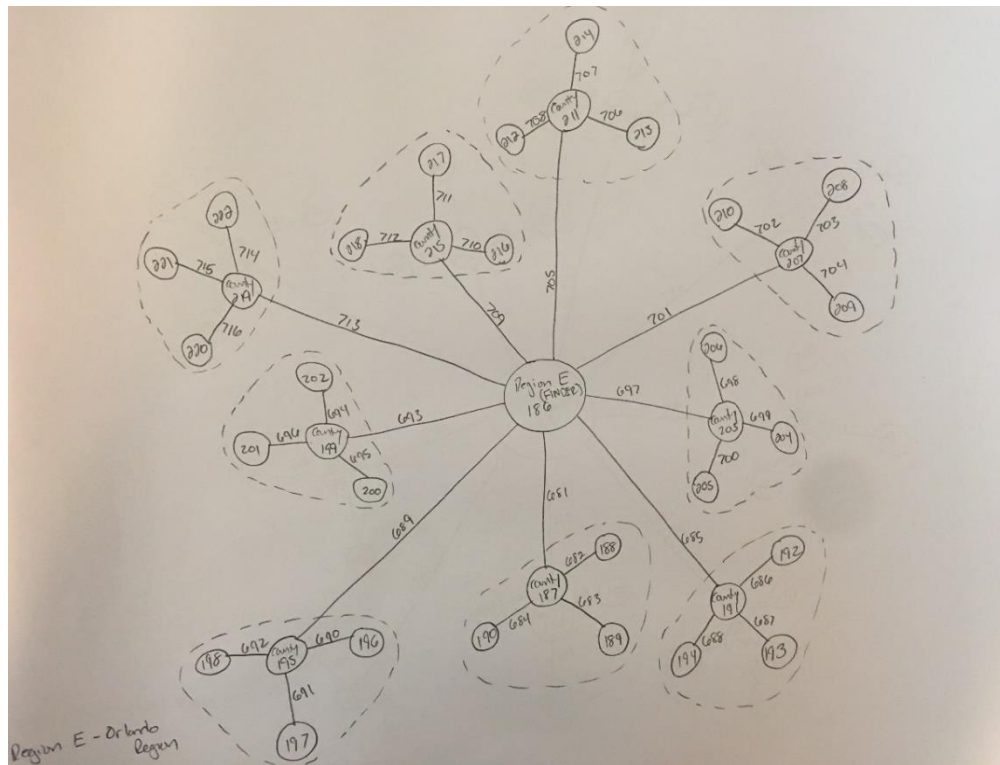
for data inputting simple. The limitations are not stoppages but rather hurdles along the way that restricted the development of the desired outcome.

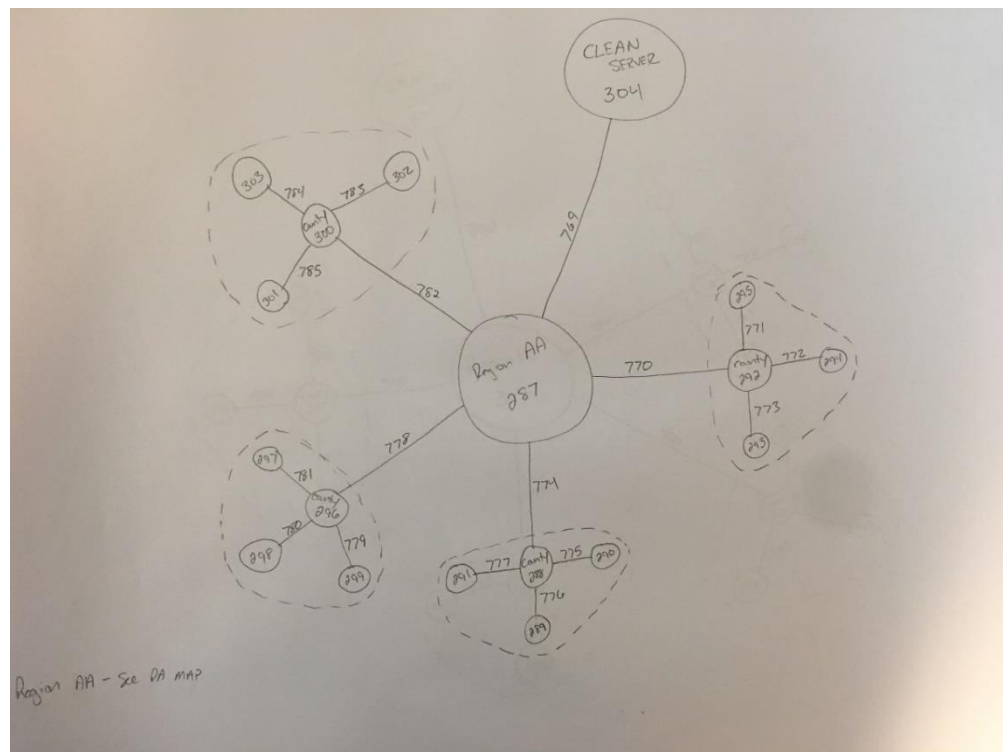
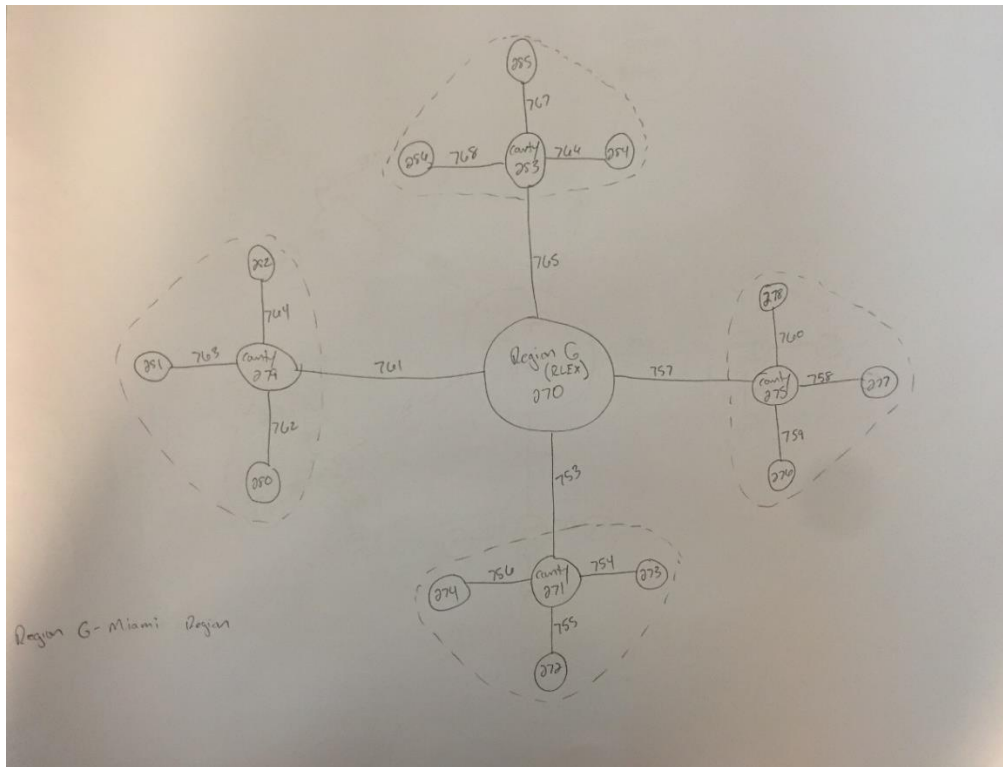
APPENDIX A. HAND SKETCHED NETWORK DIAGRAM NODE LAYOUT

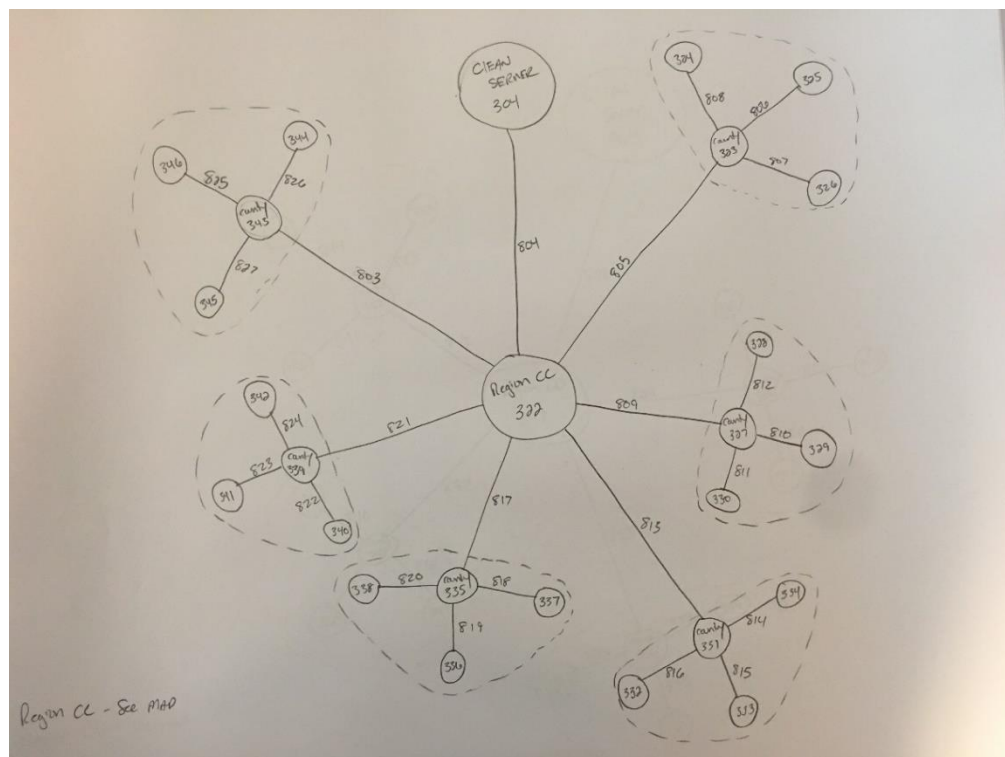
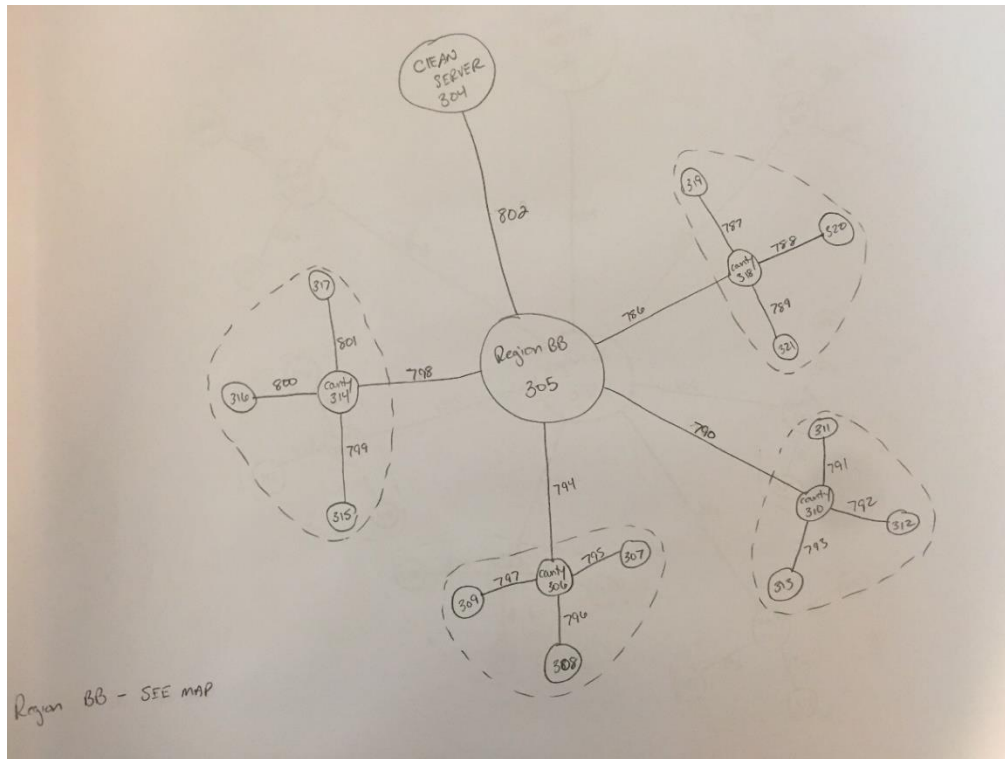


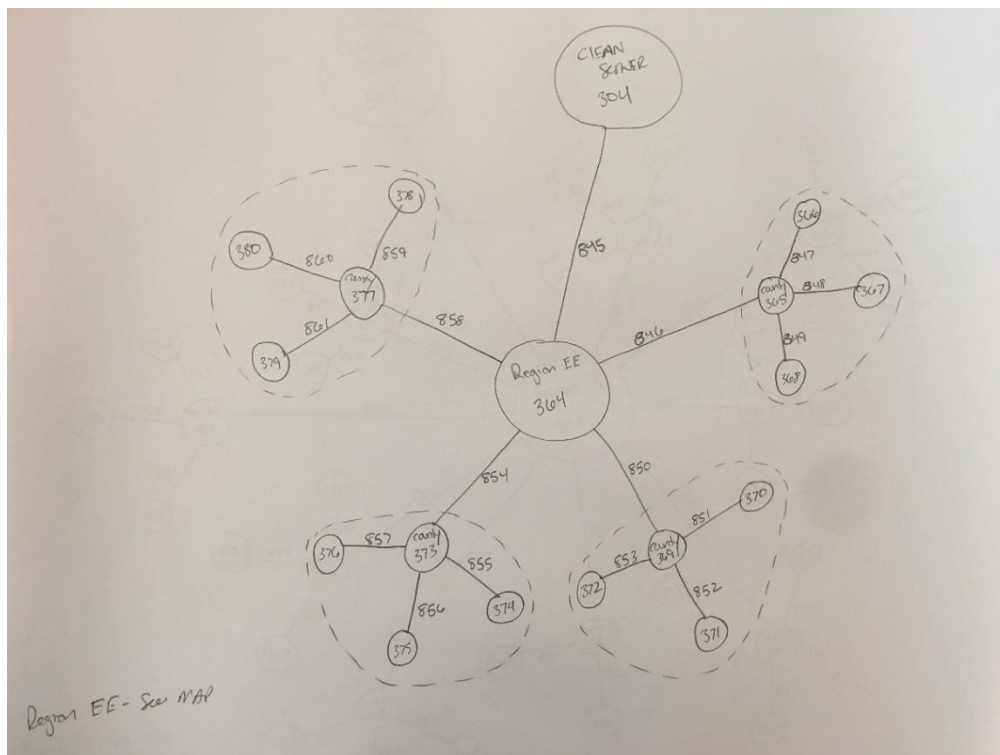
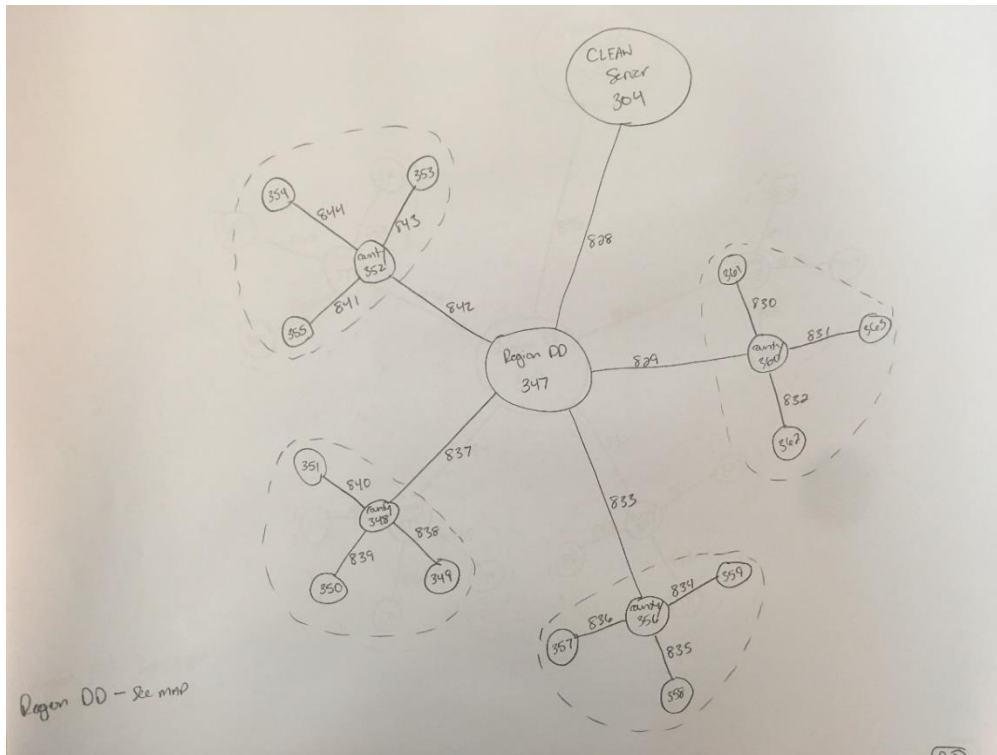


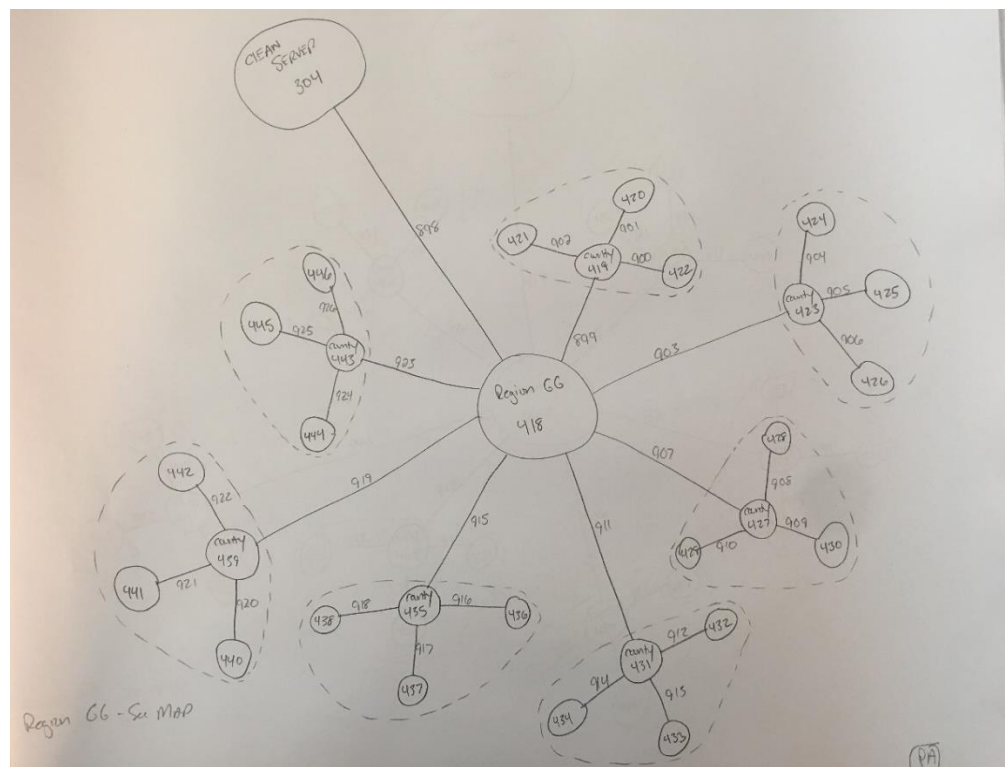
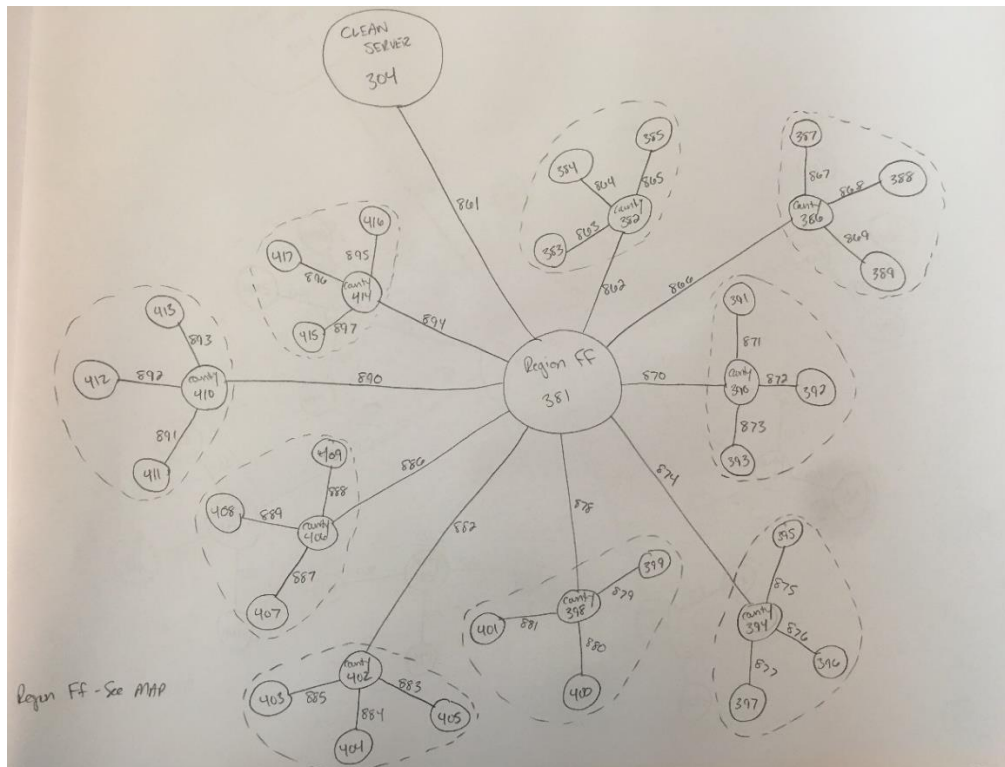


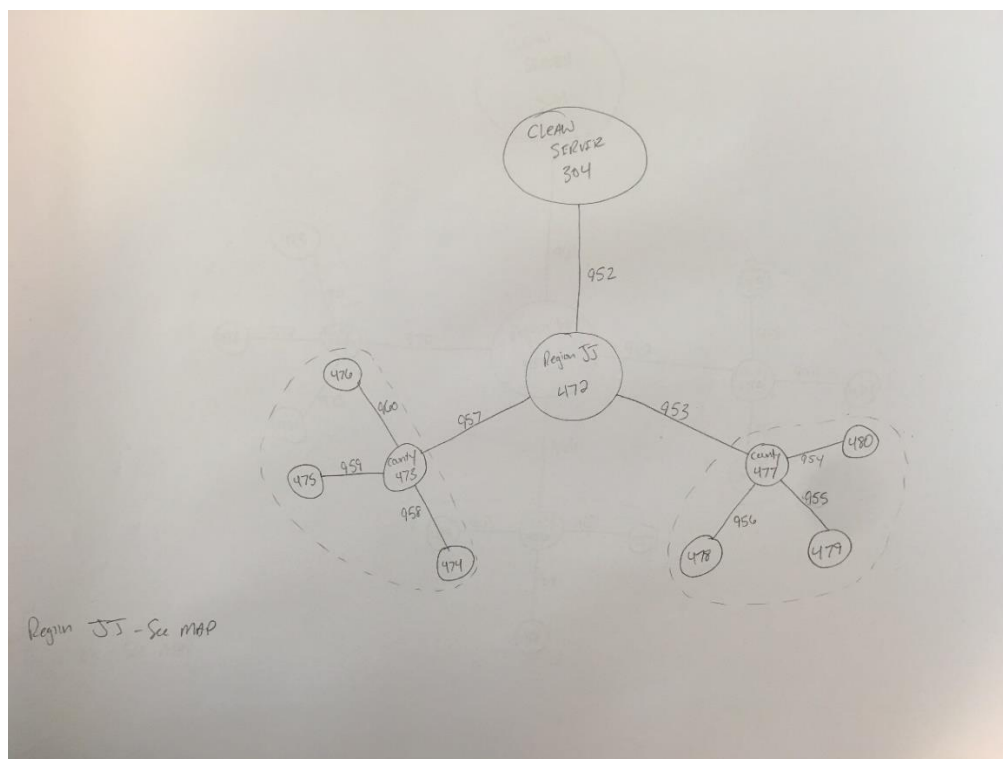
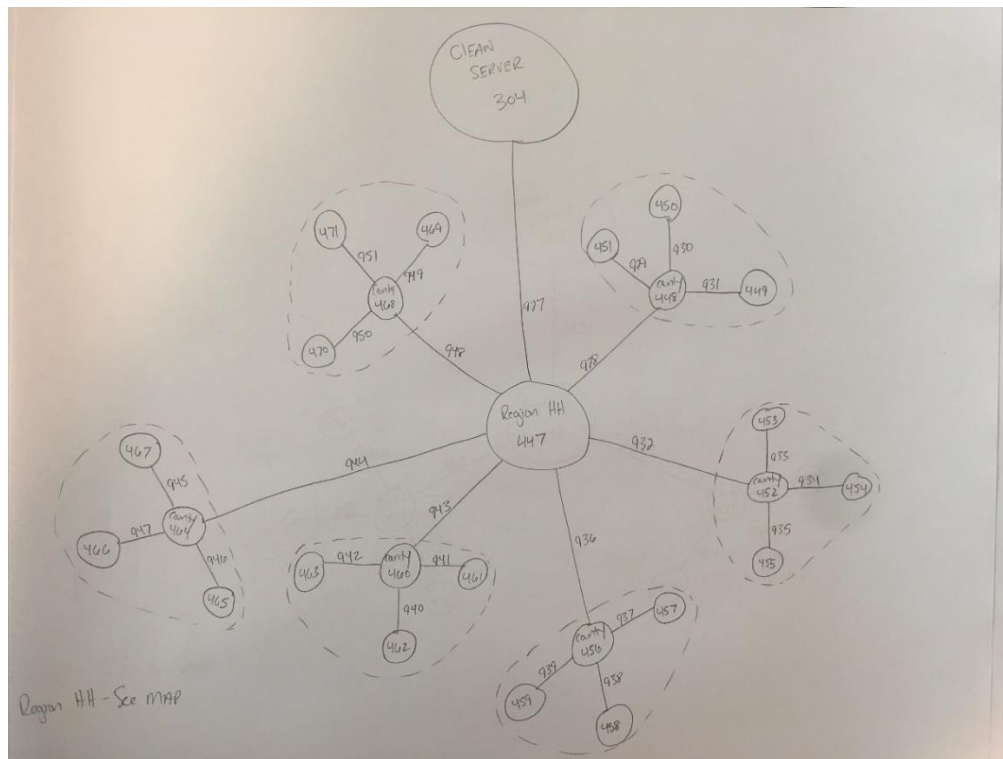


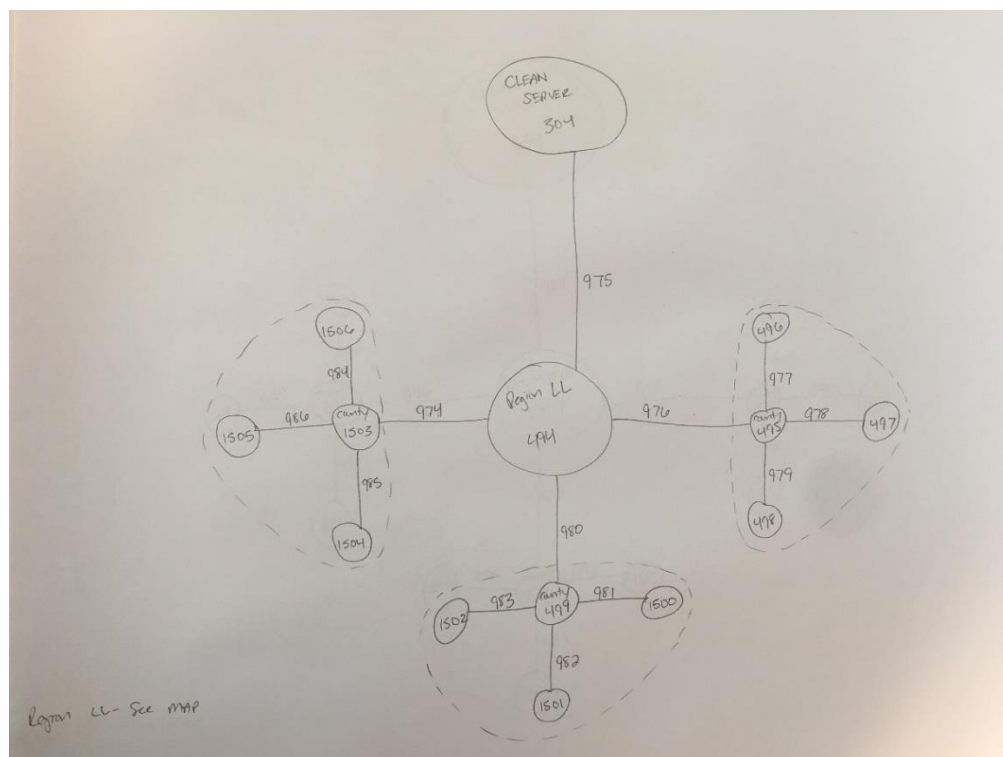
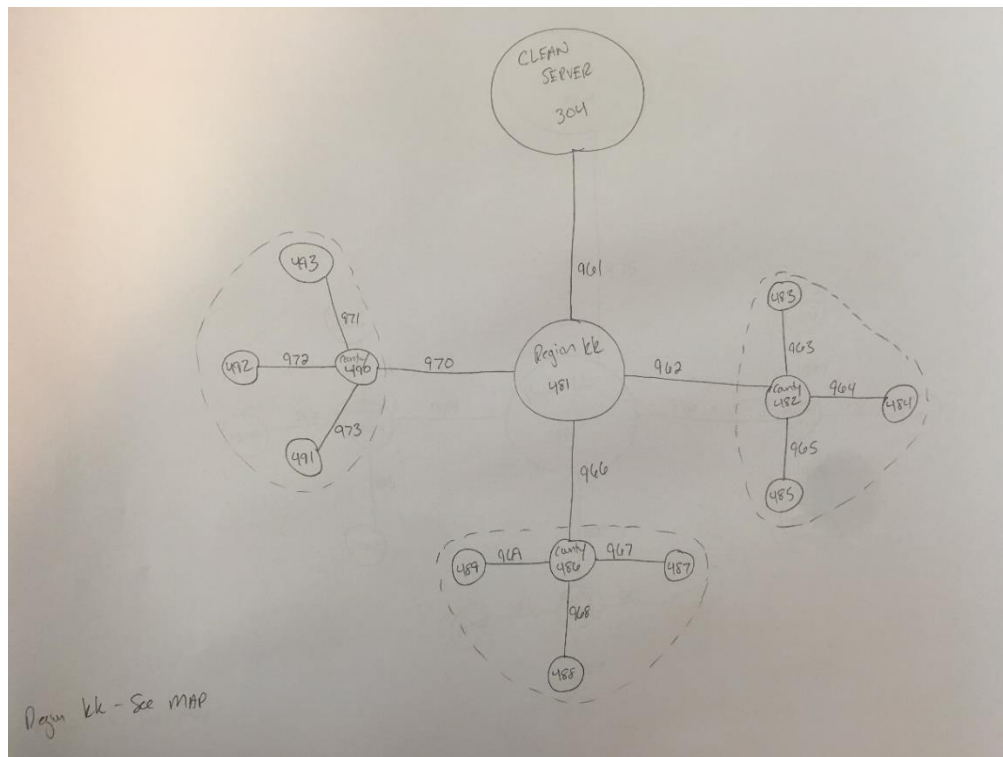


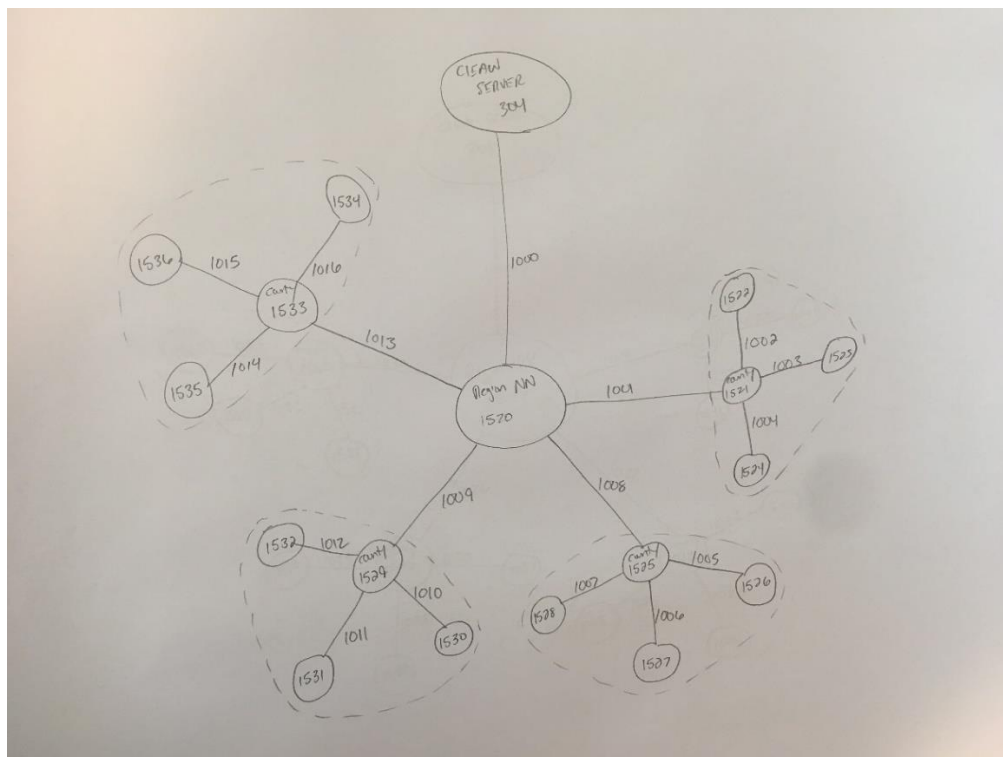
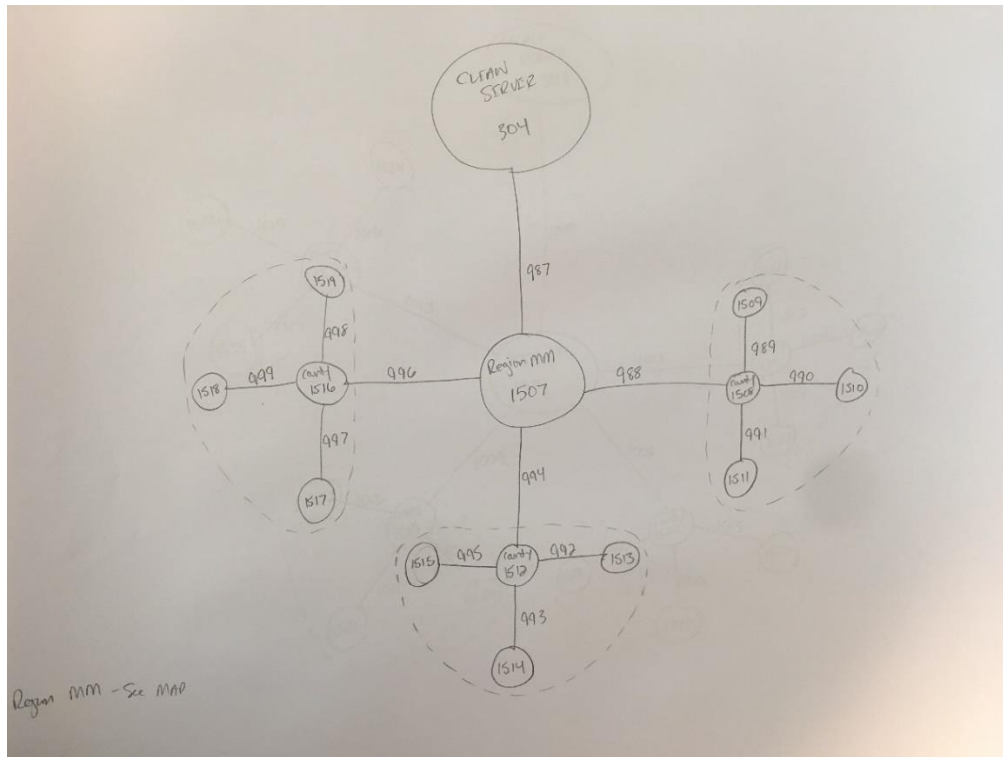


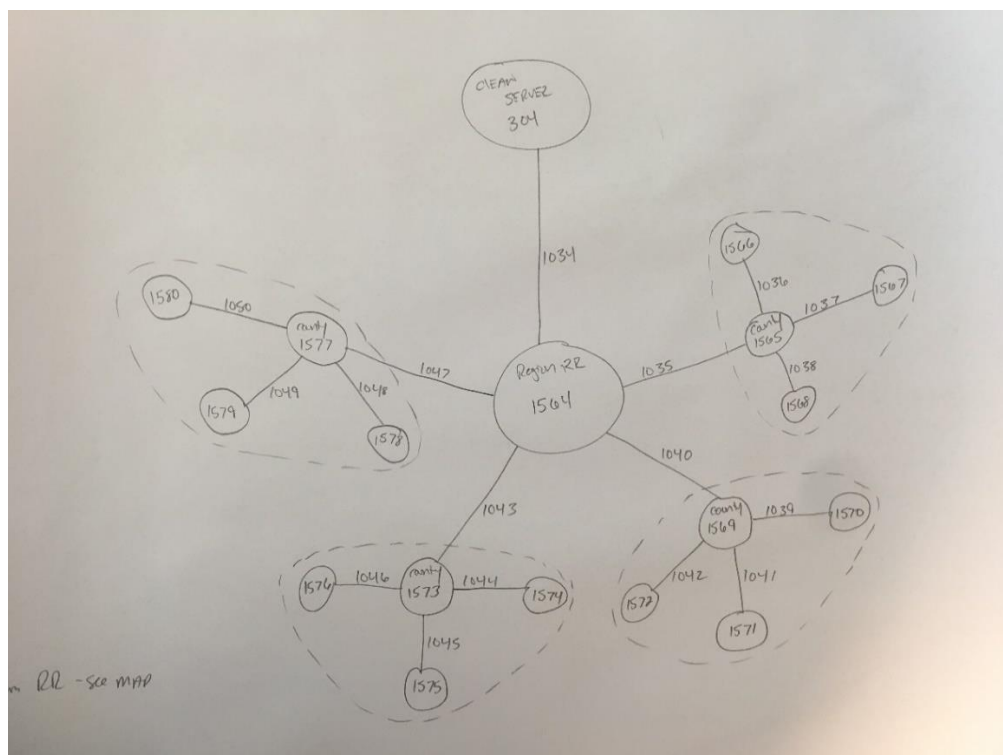
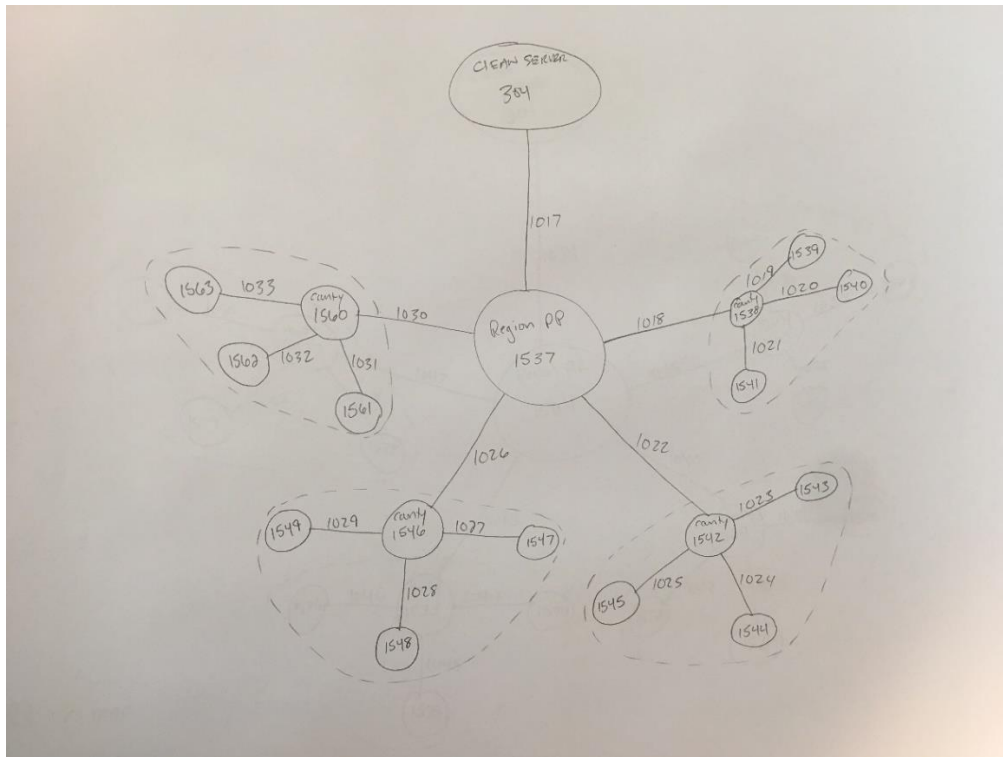






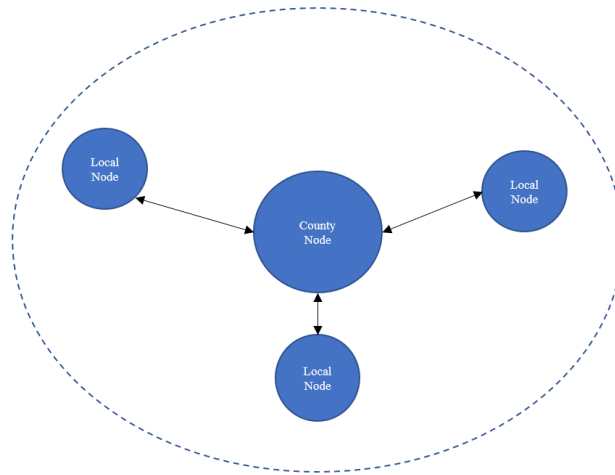




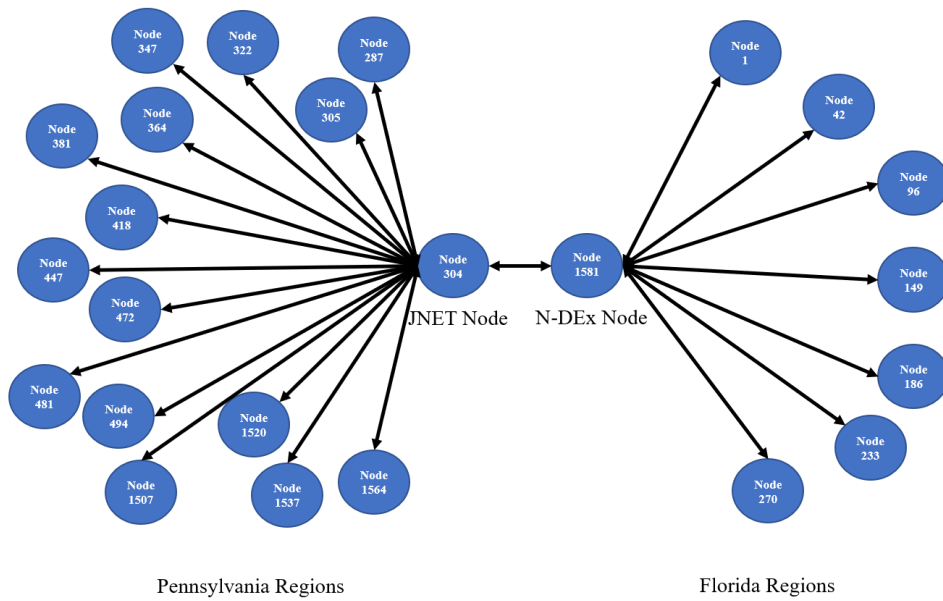


APPENDIX B. DIGITAL NODE DIAGRAMS

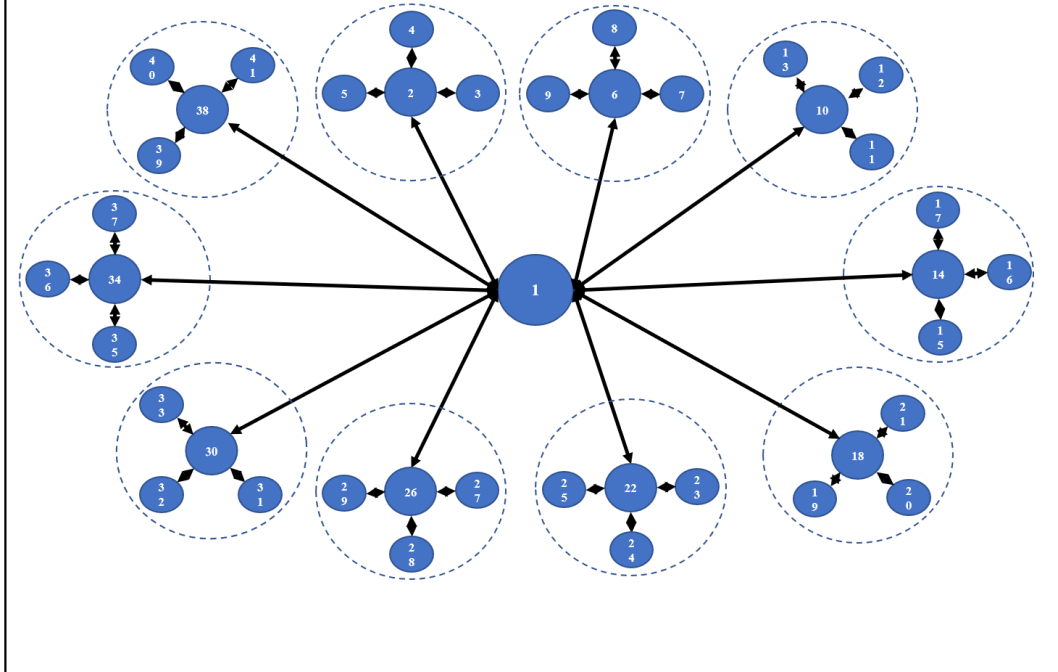
County Node Layout Example



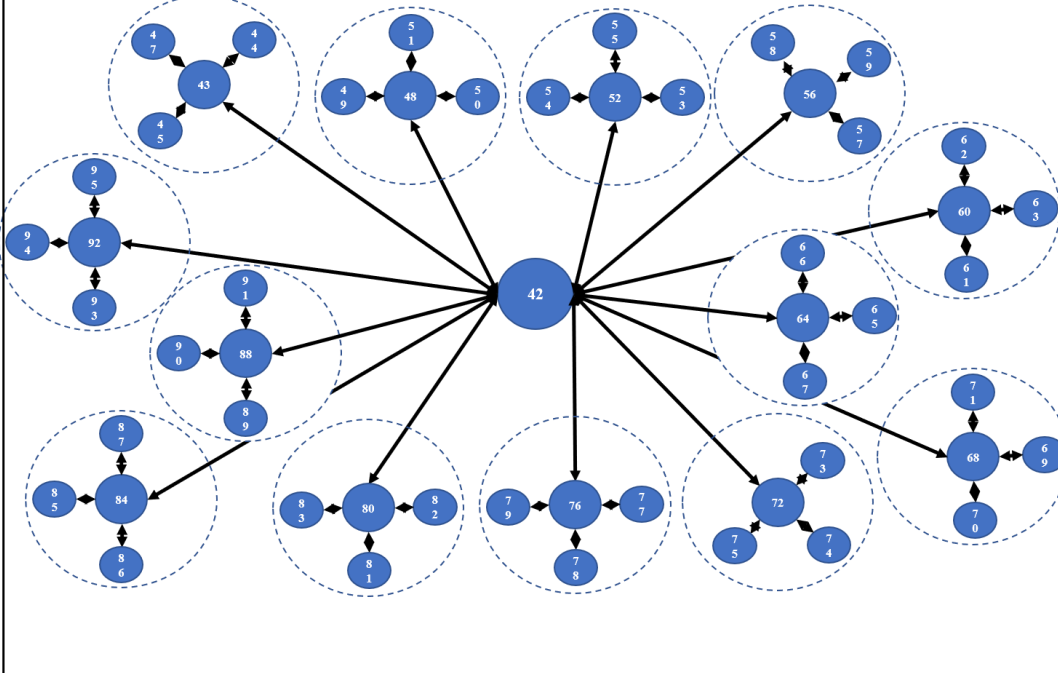
Federal Level Architecture



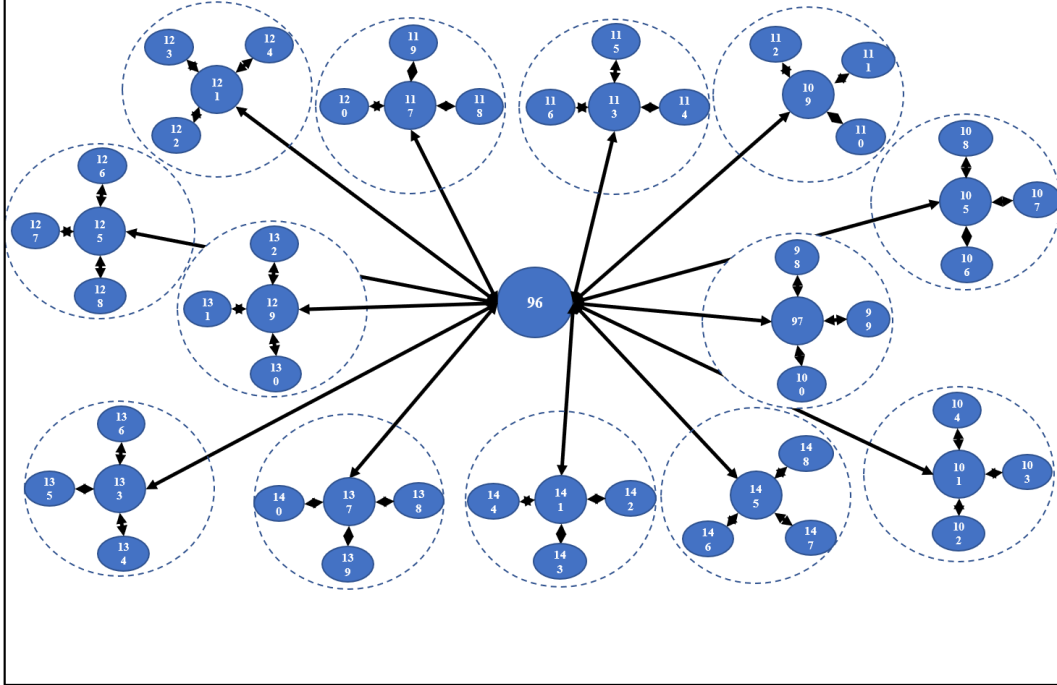
Florida Region A – Pensacola Region



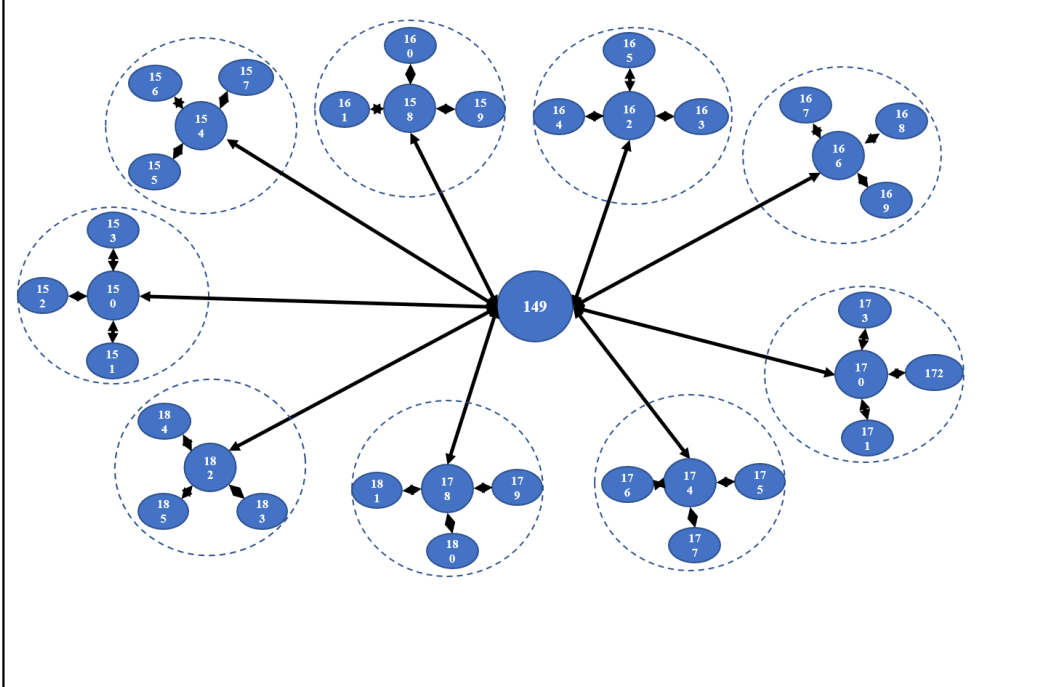
Florida Region B – Tallahassee Region



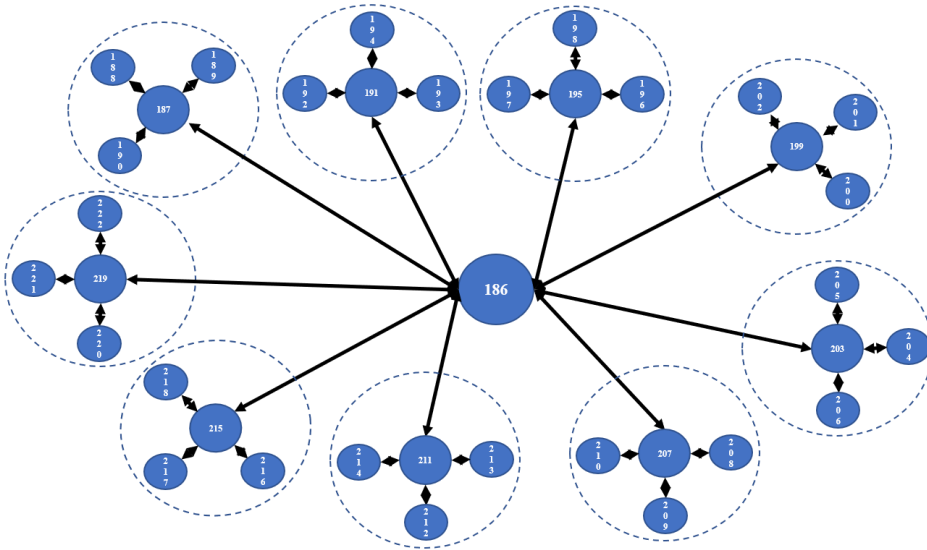
Florida Region C – Jacksonville Region



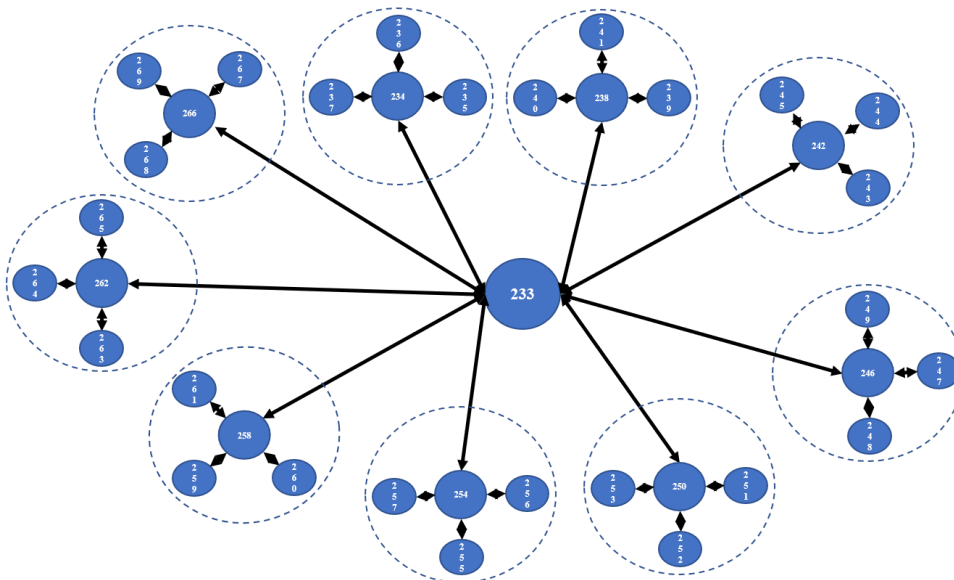
Florida Region D – Tampa Region



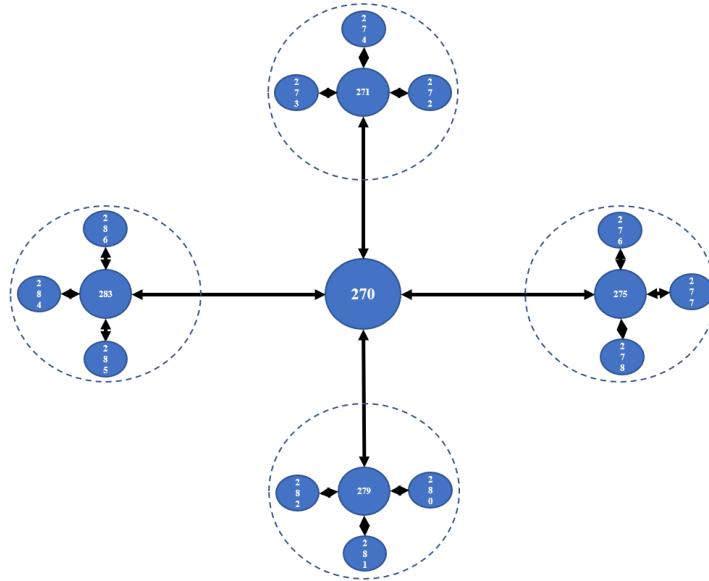
Florida Region E – Orlando Region



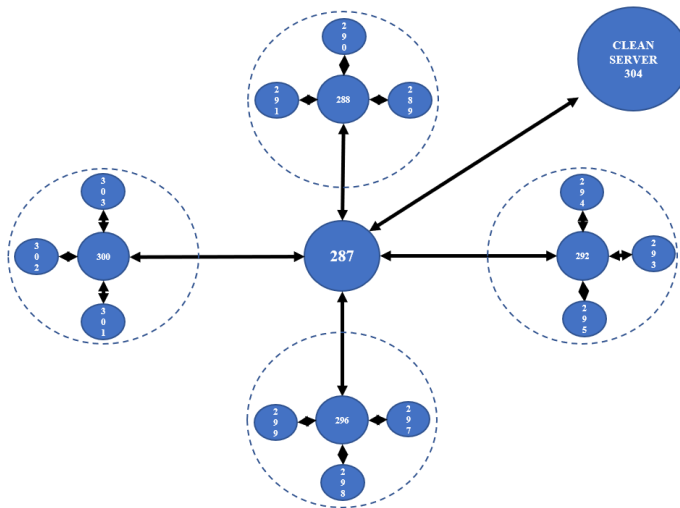
Florida Region F – Ft. Myers Region



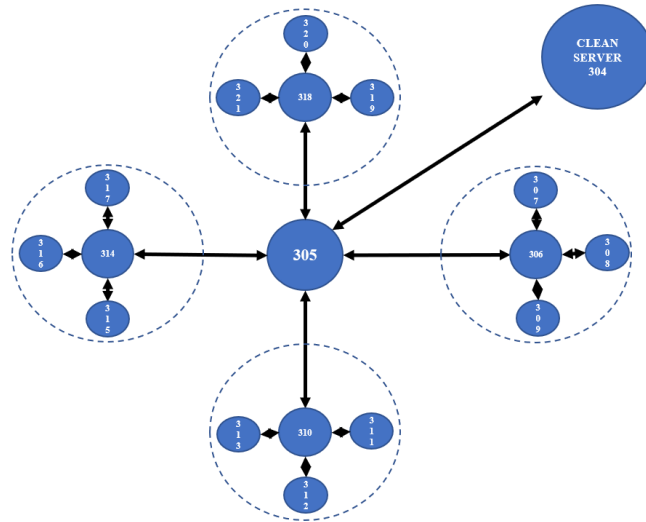
Florida Region G – Miami Region



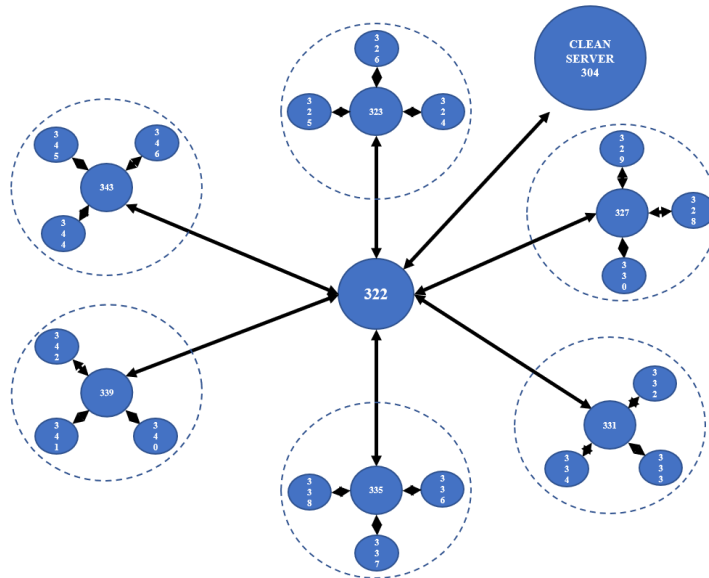
Pennsylvania Region AA



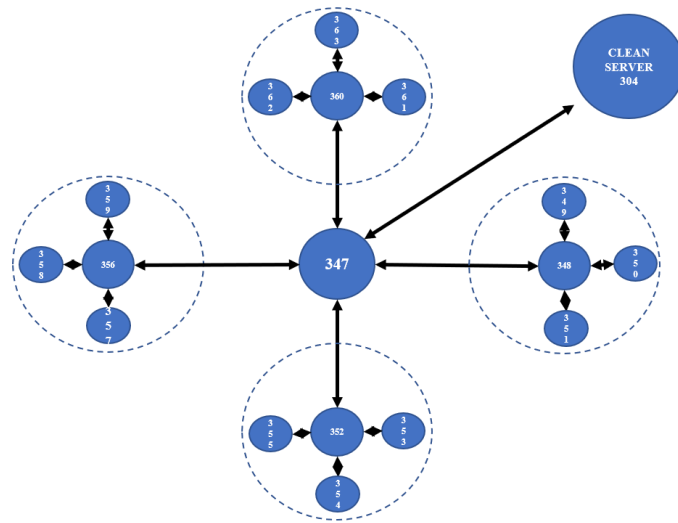
Pennsylvania Region BB



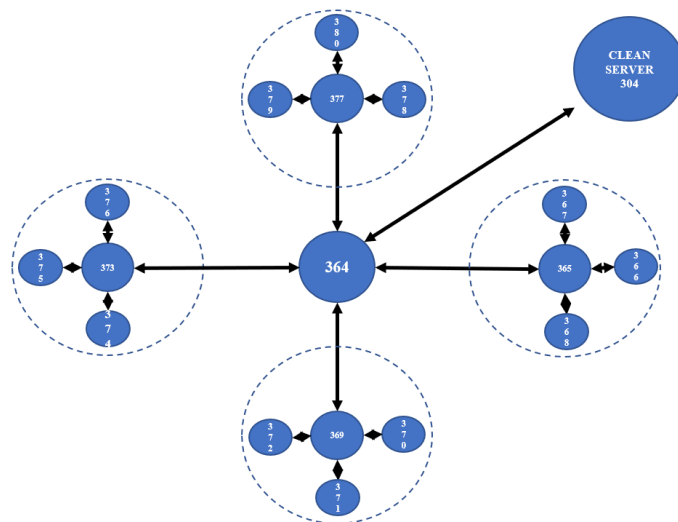
Pennsylvania Region CC

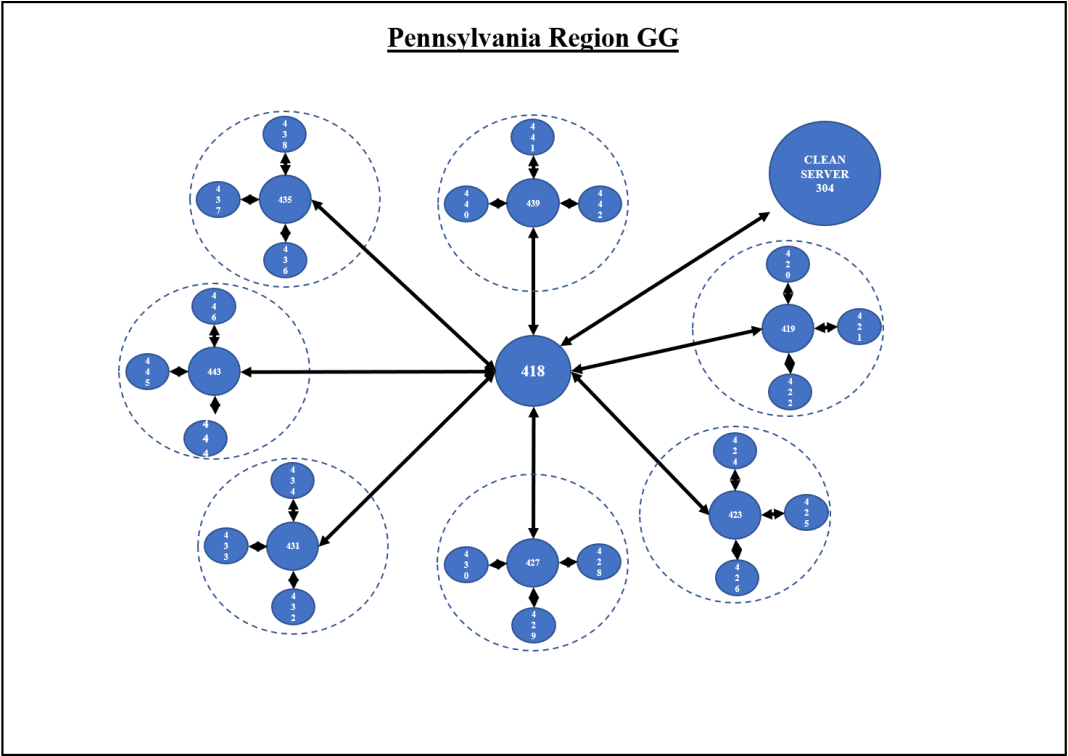
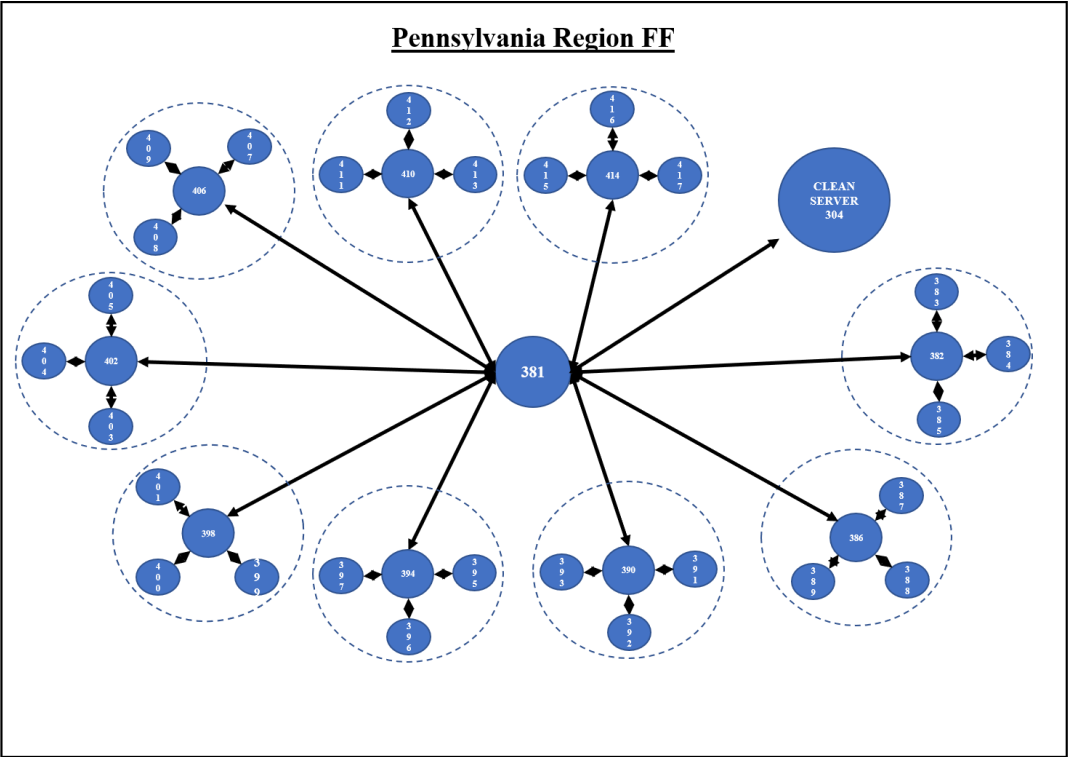


Pennsylvania Region DD

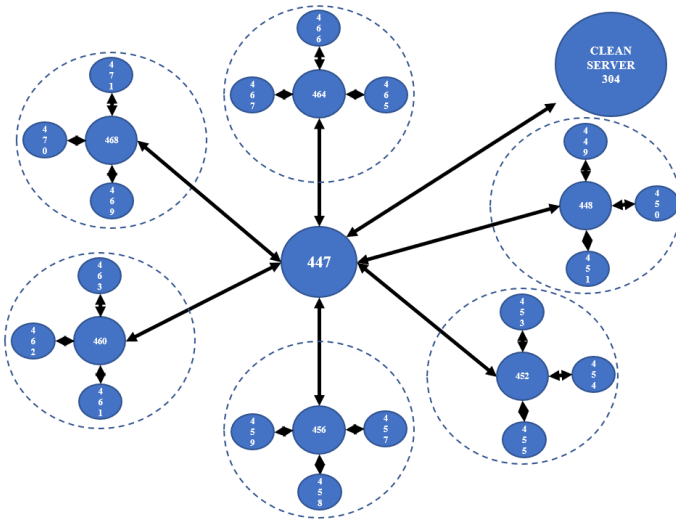


Pennsylvania Region EE

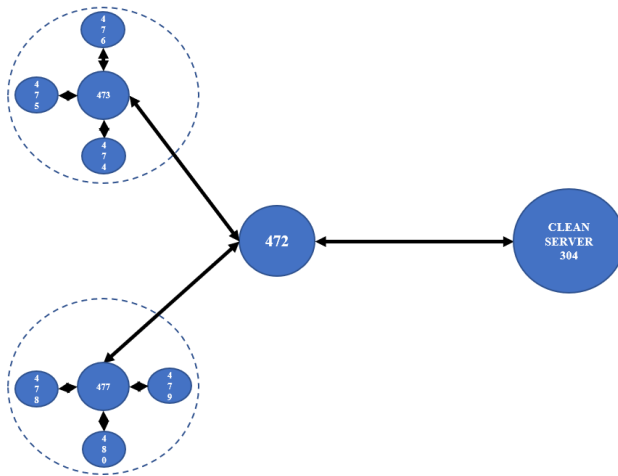




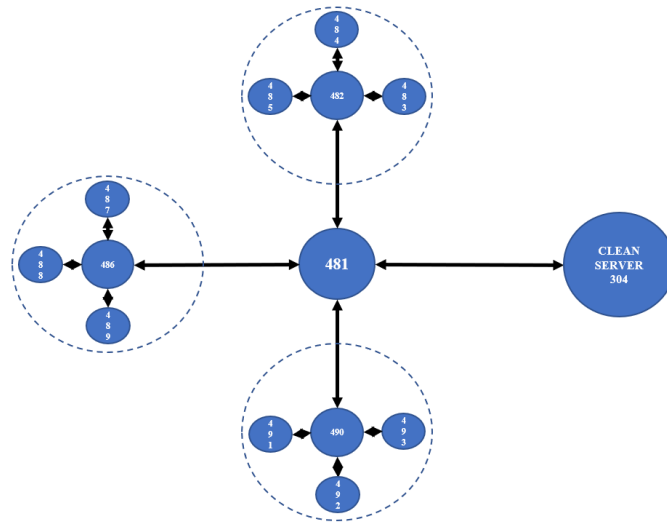
Pennsylvania Region HH



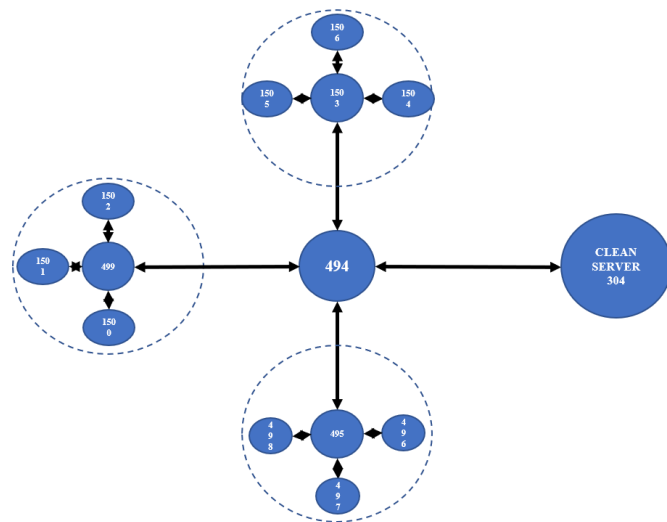
Pennsylvania Region JJ



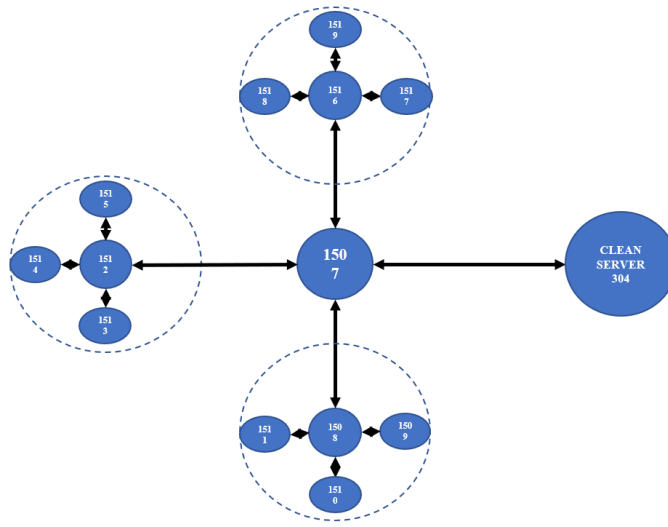
Pennsylvania Region KK



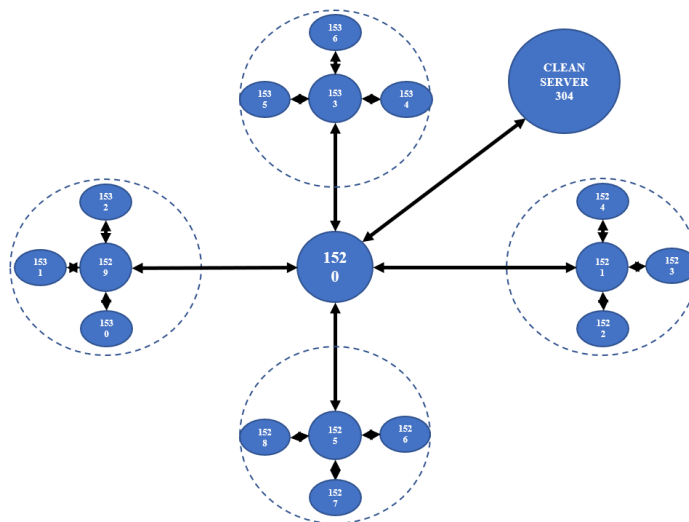
Pennsylvania Region LL



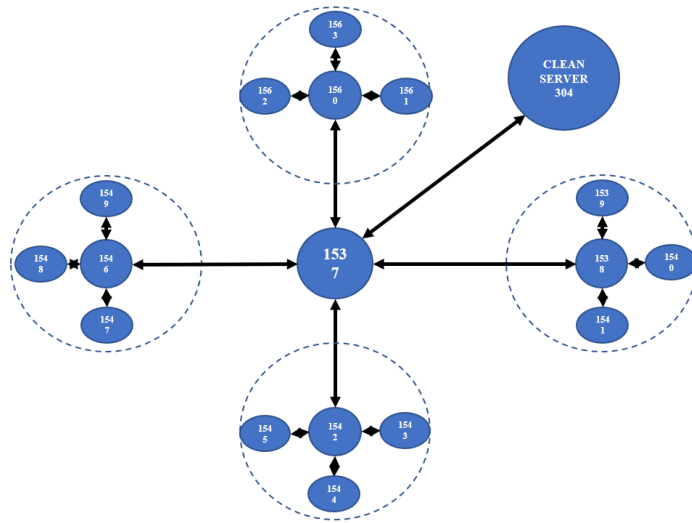
Pennsylvania Region MM



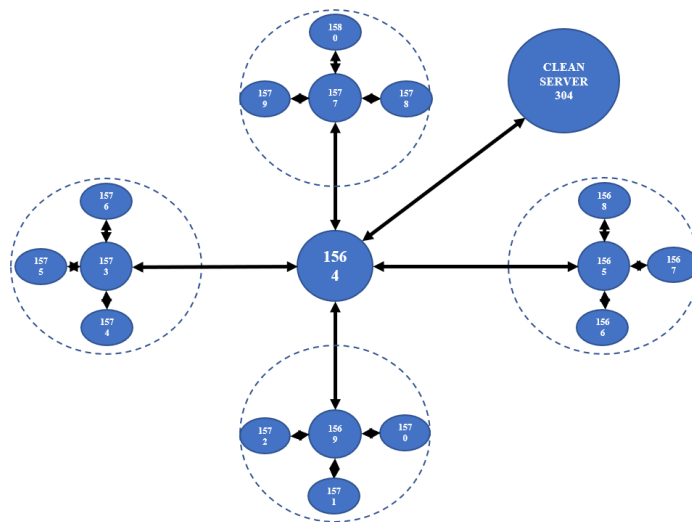
Pennsylvania Region NN



Pennsylvania Region PP



Pennsylvania Region RR



APPENDIX C. COMPLETE PTYHON CODE FOR MODEL AND SIMULATION

```

#!/usr/bin/env python
# encoding: utf-8
# Modified slightly from the gist at:
# http://timotheepoisot.fr/2012/05/18/networkx-metapopulations-python/
import networkx as nx
import numpy as np
import matplotlib.pyplot as plt
import csv

Agencies = 1 # Number of patches
Infected = 0.0001 # Probability of Infection
Susceptible = 0.01 # Probability of being Susceptible
Initial_Infection = 0.001 # Probability that a patch will be infected at the beginning
P_init_local = 0.005 # made this high to weight the edge of the graph to start

n_size = []

G = nx.Graph()
plt.figure(figsize=(20, 15))
Time = [0]
mFileName = "Florida_Law_Enforcement_Network.csv"
counter = 0

class patch:
    def __init__(self, status=0, pos=0):
        self.status = status

        self.pos = pos

    def __str__(self):
        return (str(self.status))

def LoadData(file):
    # Ty way. or the high way.
    global counter
    Stat = 1 if np.random.uniform() < Initial_Infection else 0
    Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
    p_top = patch(Stat, counter)
    node = G.add_node(p_top, time='1pm', name="N-DEx")
    counter = counter + 1

```

```

n_size.append(10000)

with open(file, newline="") as csvfile:
    reader = csv.DictReader(csvfile)
    state = ""
    oldstate = ""
    state_counter = 0
    county_counter = 0
    region_counter = 0
    local_counter = 0
    region = ""
    oldregion = ""
    county = ""
    oldcounty = ""
    local = ""
    oldlocal = ""
    p_state = ""
    p_reg = ""
    p_count = ""

    for row in reader:
        # Print the data (for debugging purposes - comment out.)
        # print(row['state'], row['state_infection'], row['region'], row['region_infection'],
row['county'],
        # row['county_infection'], row['local'], row['local_infection'])

        # This is quick and dirty just to prove you can do it.
        state = row['state']
        # state level.
        # if its a new state make a new top node.
        if state != oldstate:
            # Stat = 1 if np.random.uniform() < state_infection else 0
            Stat = 1 if np.random.uniform() < float(row['state_infection']) else 0
            Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
            p_state = patch(Stat, Pos)
            G.add_node(p_state, name=state)
            G.add_edge(p_top, p_state, weight=.25)
            state_counter = counter
            counter = counter + 1
            oldstate = state
            n_size.append(int(row['state_node_size']))

        region = row['region']
        if region != oldregion:

```

```

# Stat = 1 if np.random.uniform() < region_infection else 0
Stat = 1 if np.random.uniform() < float(row['region_infection']) else 0
Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
p_reg = patch(Stat, Pos)
G.add_node(p_reg, name=region)
G.add_edge(p_state, p_reg, weight=.25)
region_counter = counter
counter = counter + 1
oldregion = region
n_size.append(int(row['region_node_size']))

county = row['county']
if county != oldcounty:
    # Stat = 1 if np.random.uniform() < county_infection else 0
    Stat = 1 if np.random.uniform() < float(row['county_infection']) else 0
    Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
    p_count = patch(Stat, Pos)
    G.add_node(p_count, name=county)
    G.add_edge(p_reg, p_count, weight=.25)
    county_counter = counter
    counter = counter + 1
    oldcounty = county
    n_size.append(int(row['county_node_size']))

local = row['local']
if local != oldlocal:
    Stat = 1 if np.random.uniform() < float(row['local_infection']) else 0
    Pos = (np.random.uniform() * 10 - 5, np.random.uniform() * 10 - 5)
    # local level
    p_loc = patch(Stat, Pos)
    G.add_node(p_loc, name=local)
    G.add_edge(p_count, p_loc, weight=.25)
    local_counter = counter
    counter = counter + 1
    n_size.append(int(row['local_node_size']))

def Status():
    for n in G.nodes():
        return n

def Simulate(Infection):
    for timestep in range(500):

```

```

Status()
## Check for infections
for n in G.nodes():
    if n.status == 1 and np.random.uniform() < Infected:
        n.status = 1
## Check for Agencies that are infected
for n in G.nodes():
    if n.status == 0:
        neighb = G[n] # That's it, a list of the neighbors
        for nei in neighb:
            if nei.status == 0:
                if np.random.uniform() < Susceptible:
                    nei.status = 1
                    break

Time.append(timestep + 1)
Infection.append(np.sum([n.status for n in G]) / float(Agencies))

def main():
    print("Loading Data from file.... ", mFileName)
    LoadData(mFileName)
    pos = { }
    for n in G.nodes():
        pos[n] = n.pos
    occup = [n.status for n in G]

    degree = nx.degree_centrality(G)
    print(degree)
    bw_centrality = nx.betweenness_centrality(G, weight=10)
    print(bw_centrality)

    plt.title('Initial Infection Status', fontsize=30)
    nx.draw(G, center=1581, node_size=(n_size), node_color=occup, with_labels=False,
    cmap=plt.cm.plasma,
        vmin=0, vmax=1)

plt.savefig('beginning_infection/Florida_Law_Enforcement_network_start_infection_01[25_75].
png')
plt.show()

Infection = [np.sum([n.status for n in G]) / float(Agencies)]
Simulate(Infection)

plt.figure(figsize=(20, 15))

```

```

plt.title('Post Simulation Infection Status', fontsize=30)
nx.draw(G, center=1581, node_size=(n_size), node_color=[n.status for n in G],
with_labels=False,
        cmap=plt.cm.plasma, vmin=0, vmax=1)

plt.savefig('end_infection/Florida_Law_Enforcement_network_end_infection_01[25_75].png')
plt.show()

# line plot
plt.title('Node Infection over Time', fontsize=15)
plt.plot(Time, Infection)
plt.xlabel('TIME')
plt.ylabel('INFECTED')
# Customize the major grid
plt.grid(which='major', linestyle='-', linewidth='0.5', color='red')
# Customize the minor grid
plt.grid(which='minor', linestyle=':', linewidth='0.5', color='black')
plt.savefig('infection_timeplot/Florida_Law_Enforcement_network_plot_01[25_75].pdf')
plt.tight_layout()
plt.show()

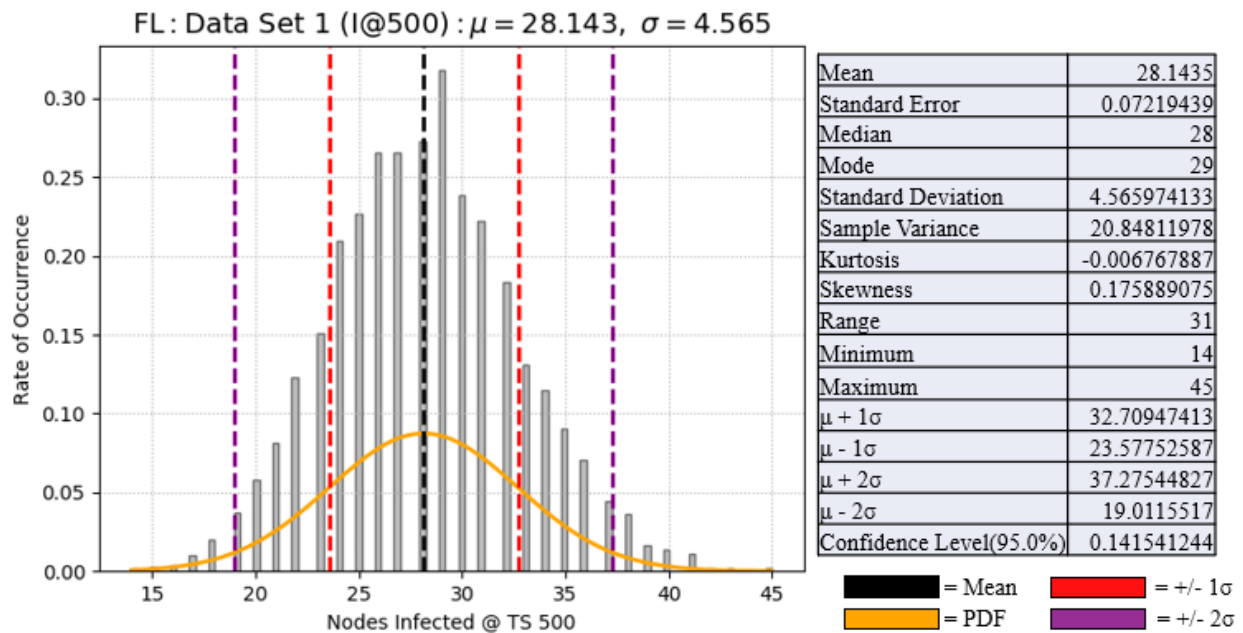
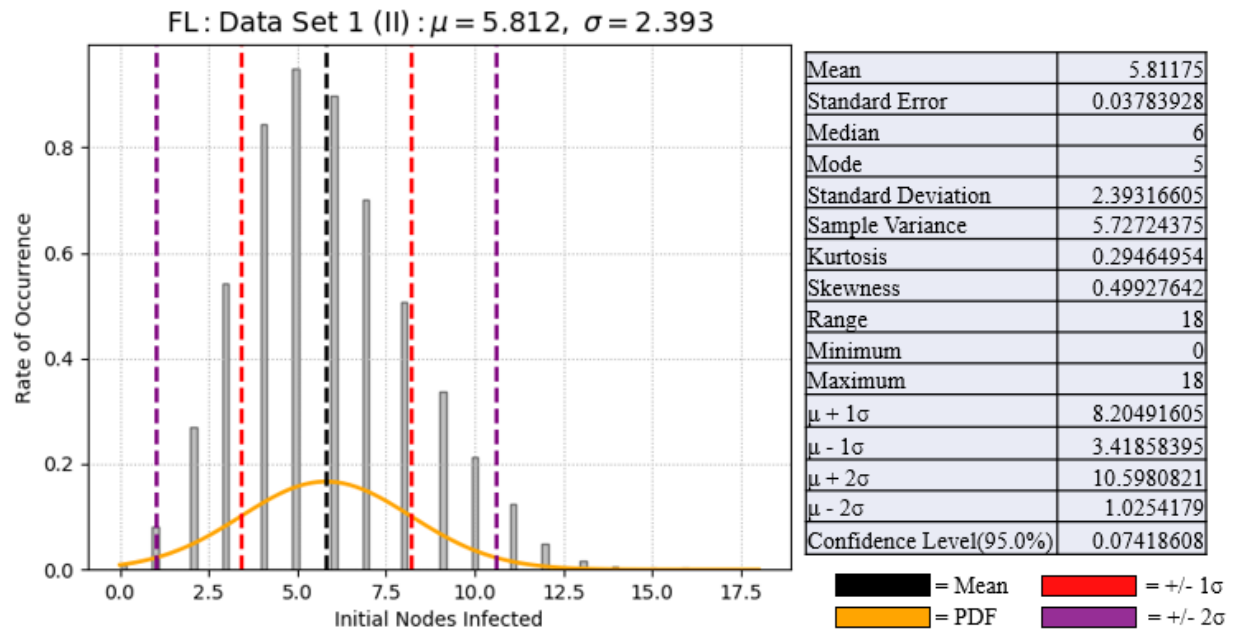
data = [np.sum([n.status for n in G]) / float(Agencies)]
print(nx.info(G))
print(nx.degree_histogram(G))
density = nx.density(G)
e1 = sum(occup)
e2 = sum(data)
e3 = (e2 - e1)/500
print("Network density:", density)
print('Initial Node Infection =', e1)
print('Nodes Infected =', e2)
print('Infection Rate =', e3)
print(e1)
print(e2)
print(e3)

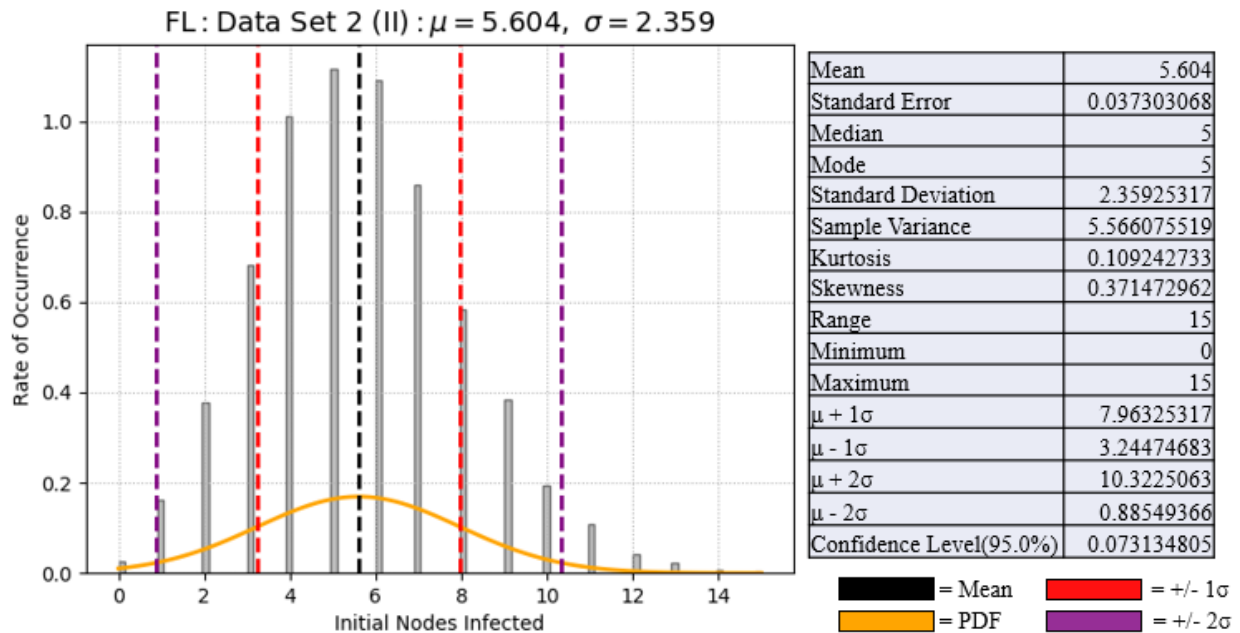
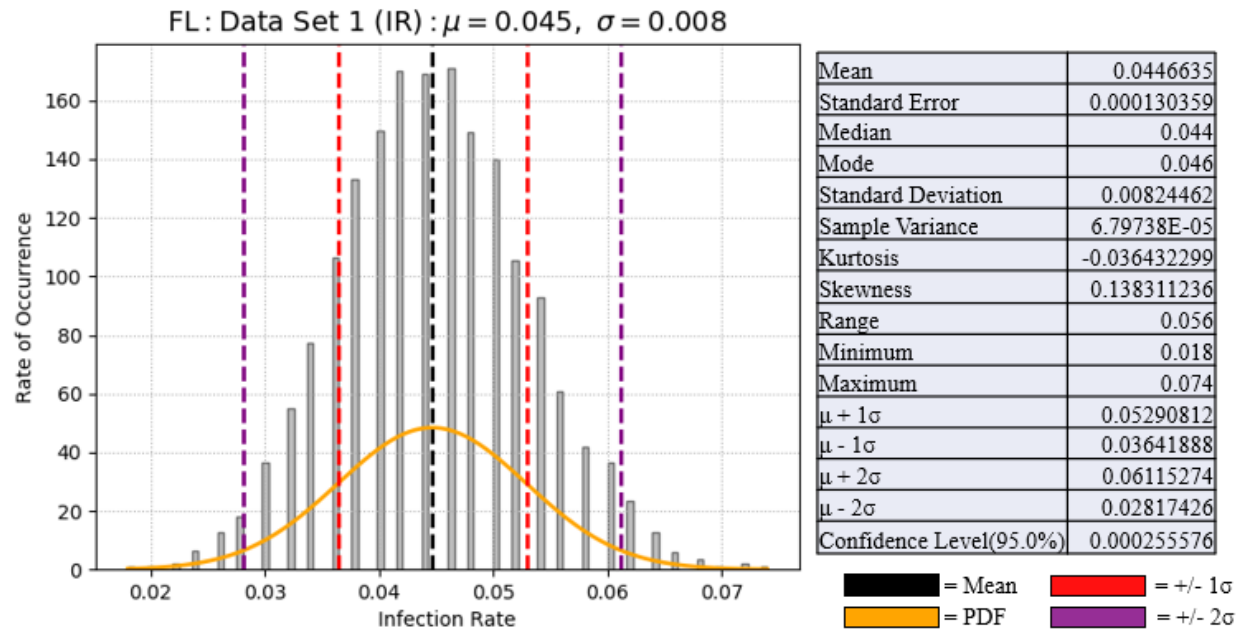
#print(e1, e2, e3, file=open("output_0001_0001_0001.txt", "a"))

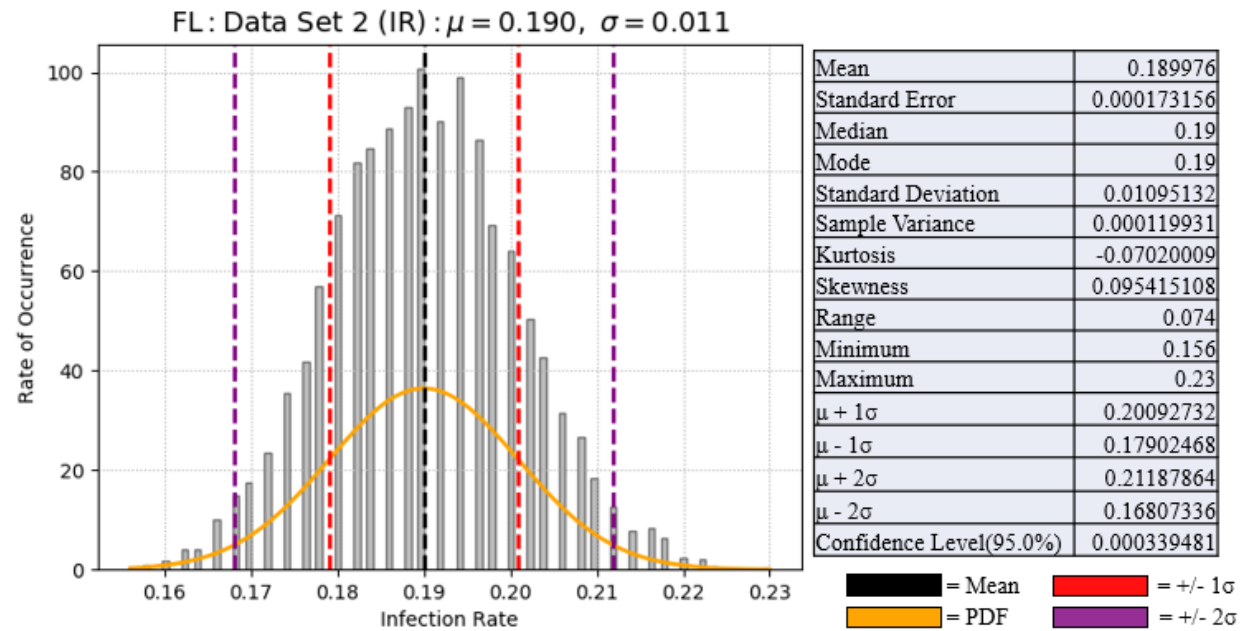
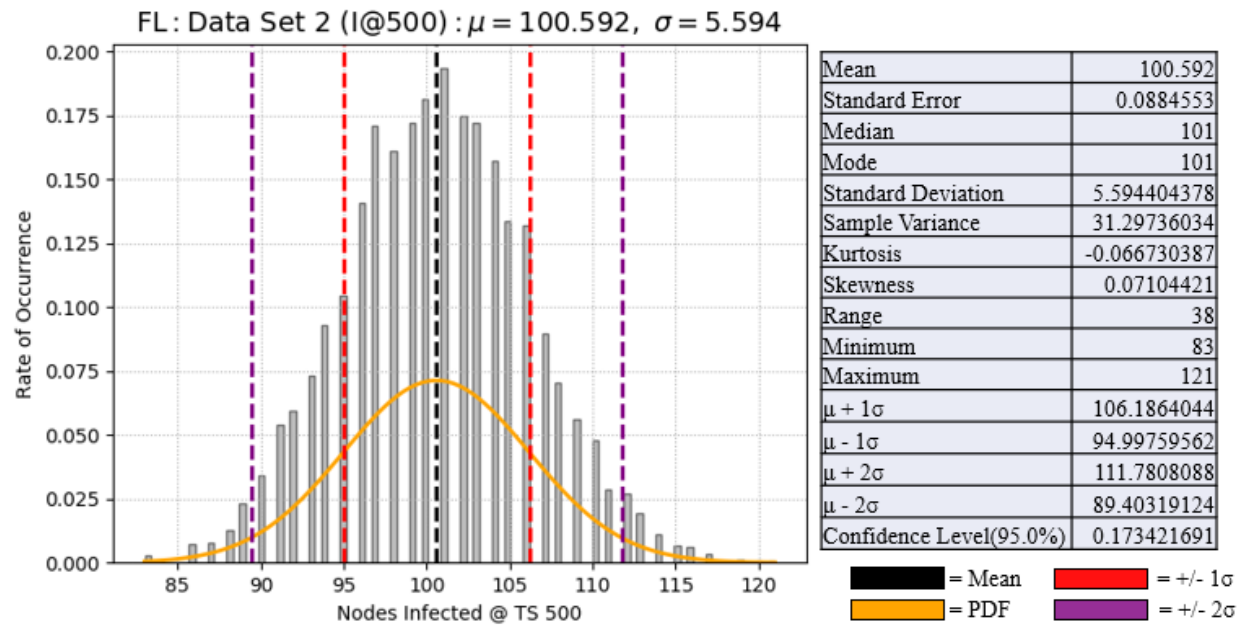
if __name__ == "__main__":
    main()

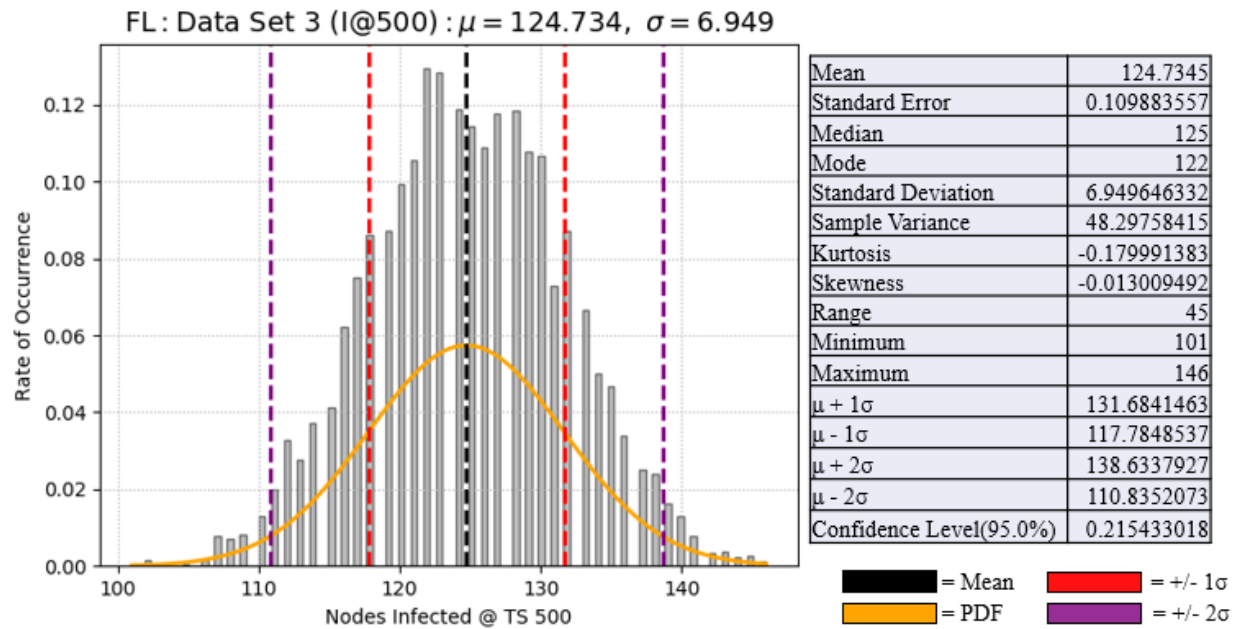
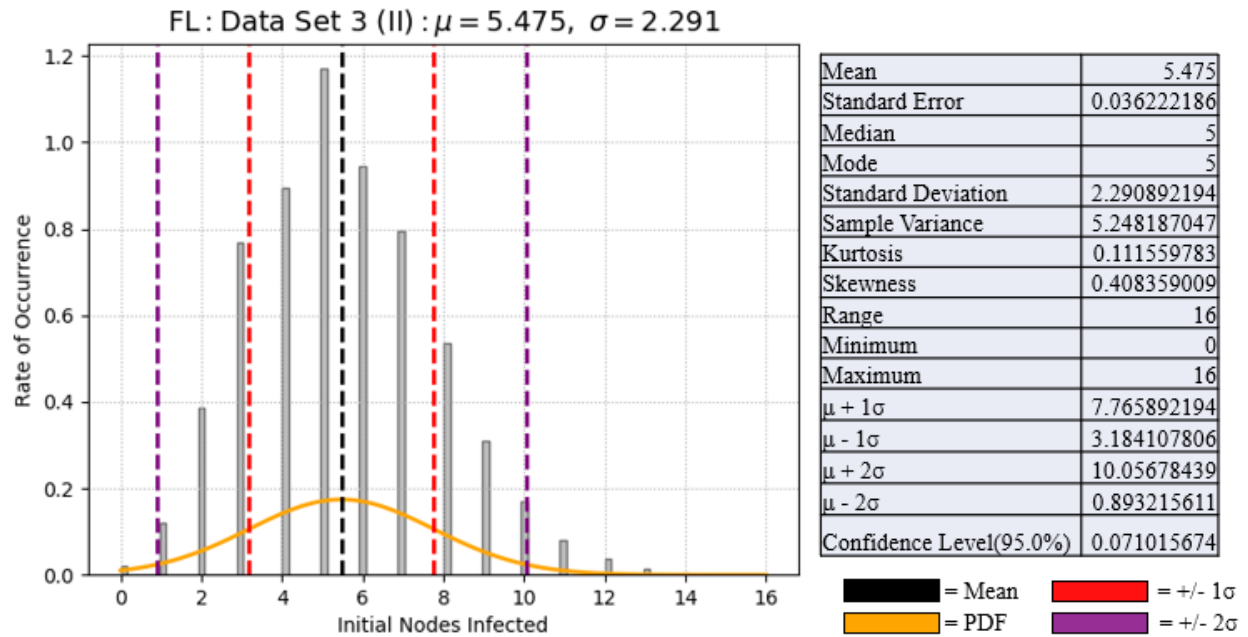
```

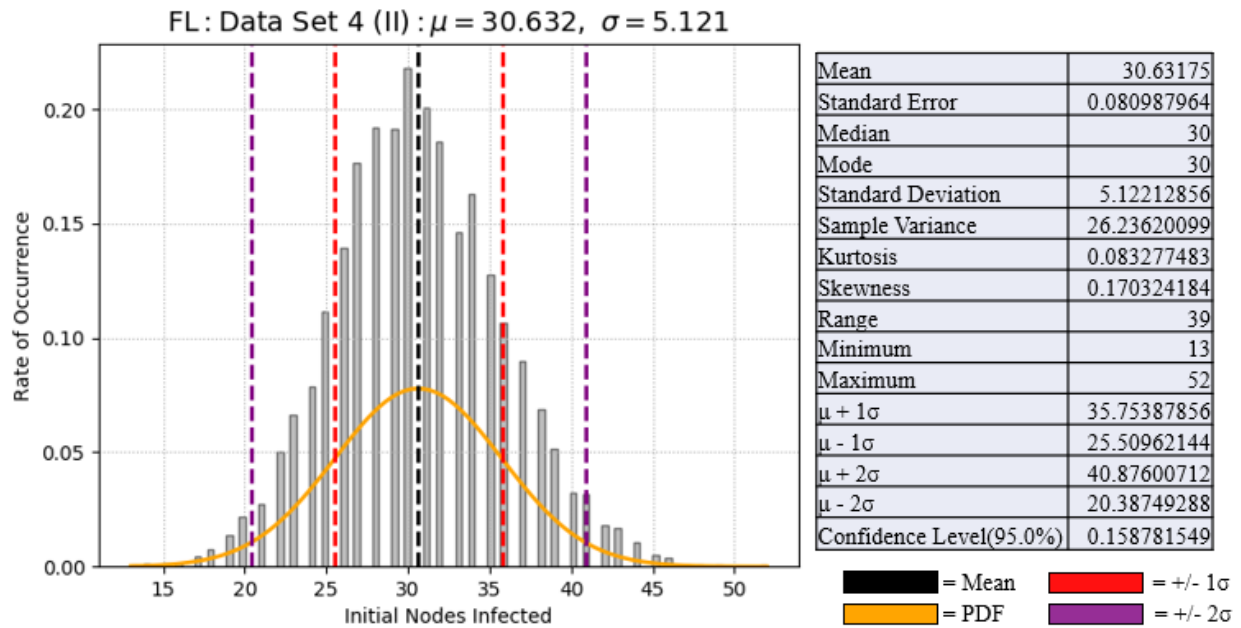
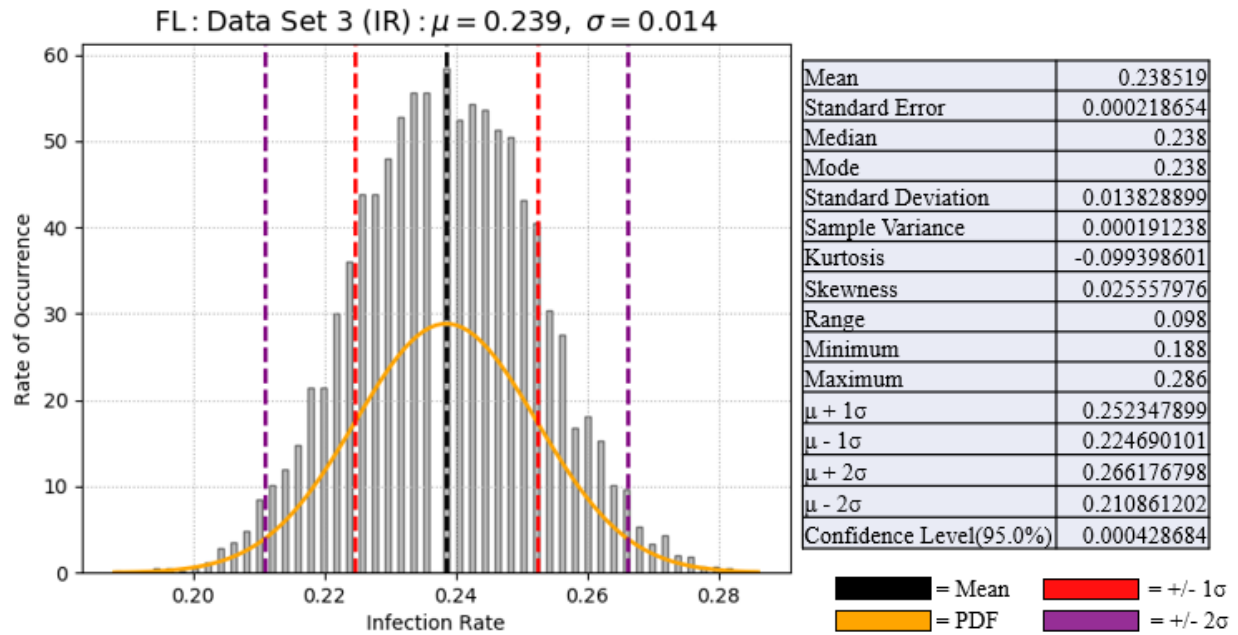
APPENDIX D. FLORIDA NETWORK DATA GRAPHS AND STATISTICS

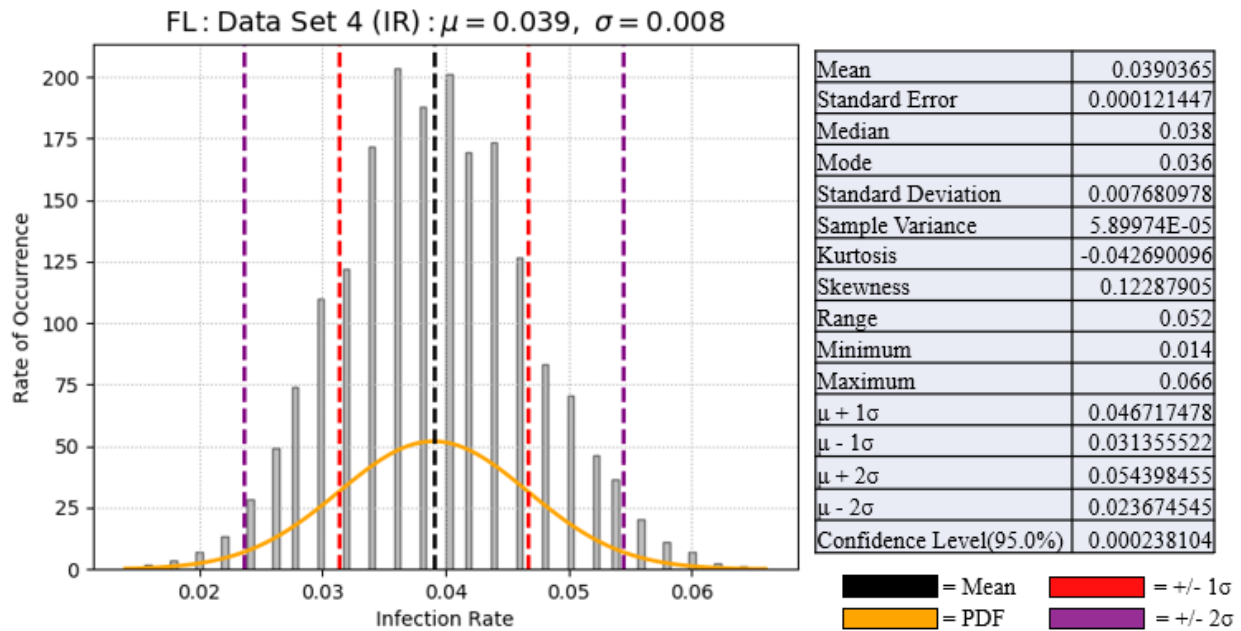
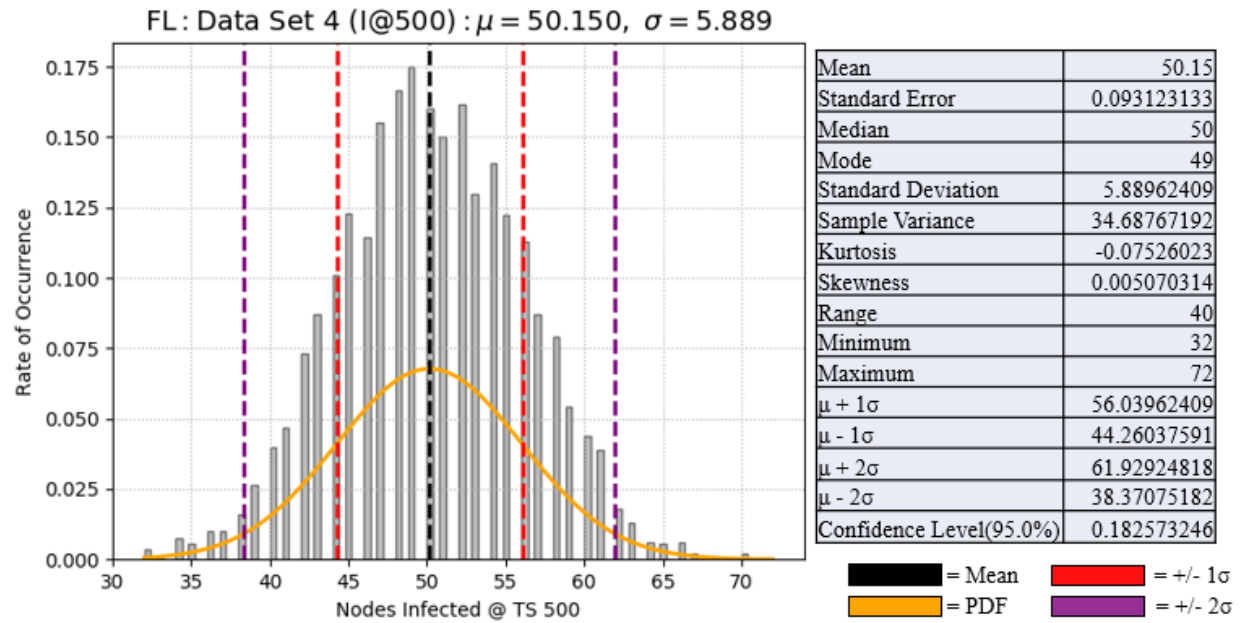


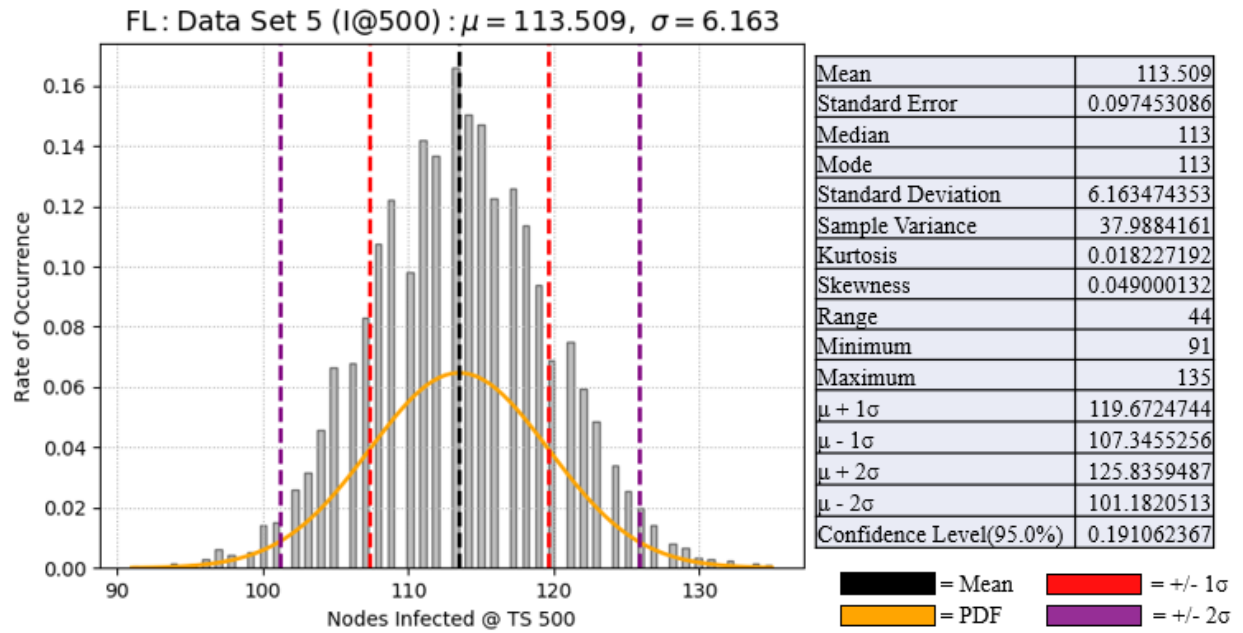
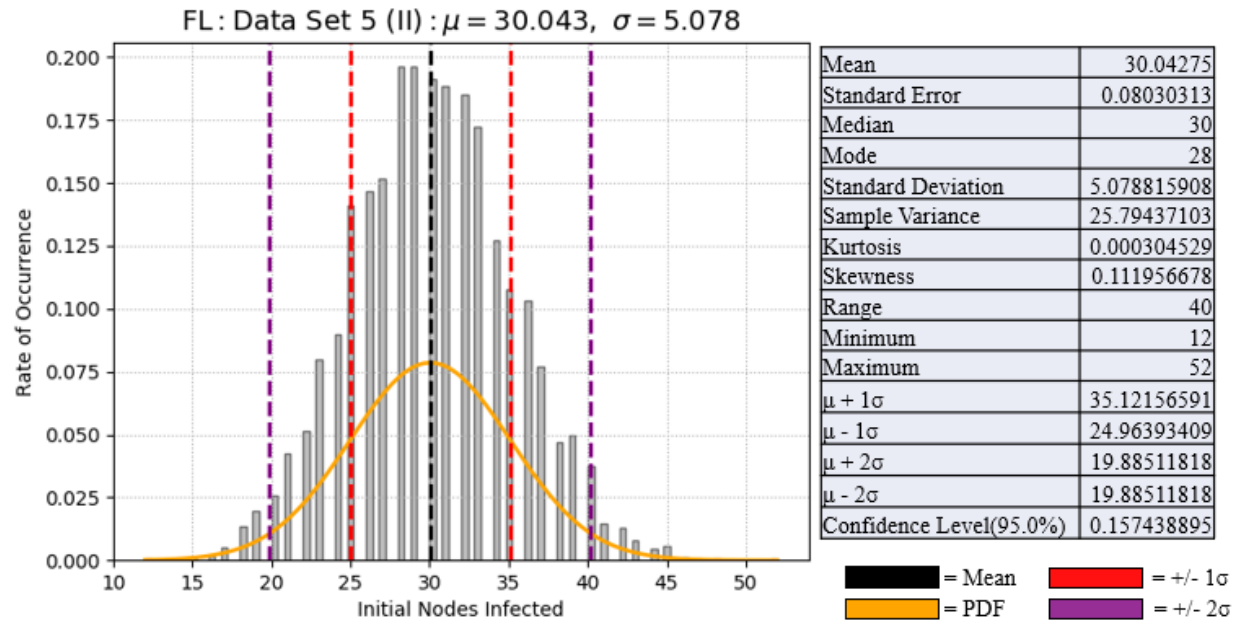


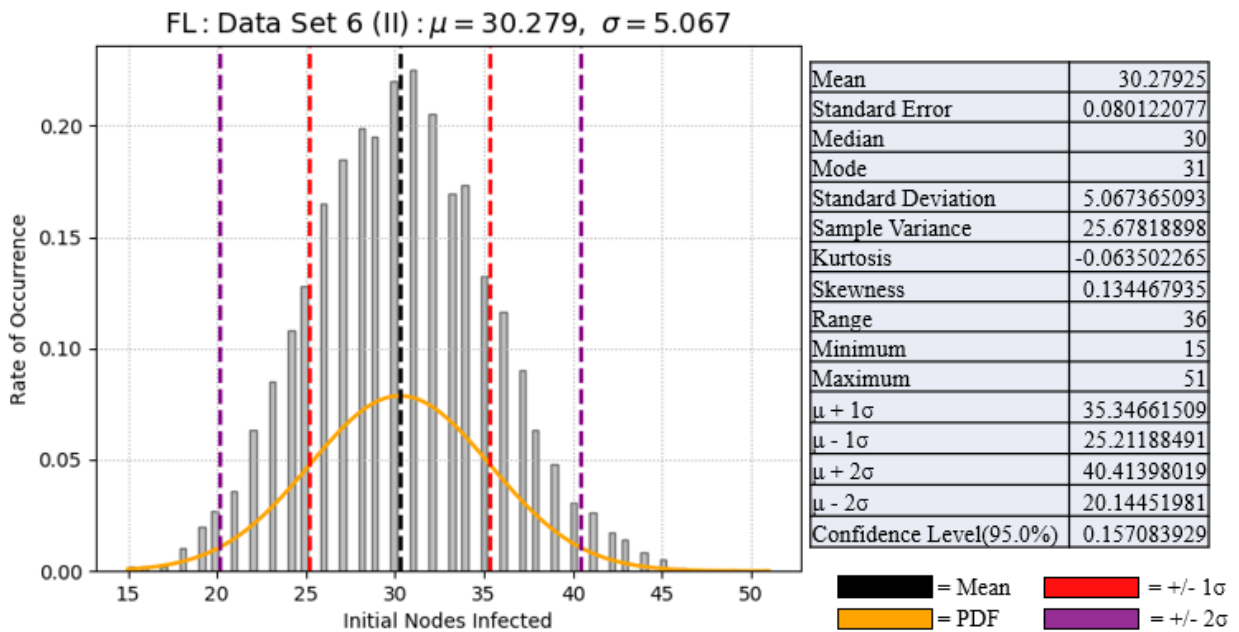
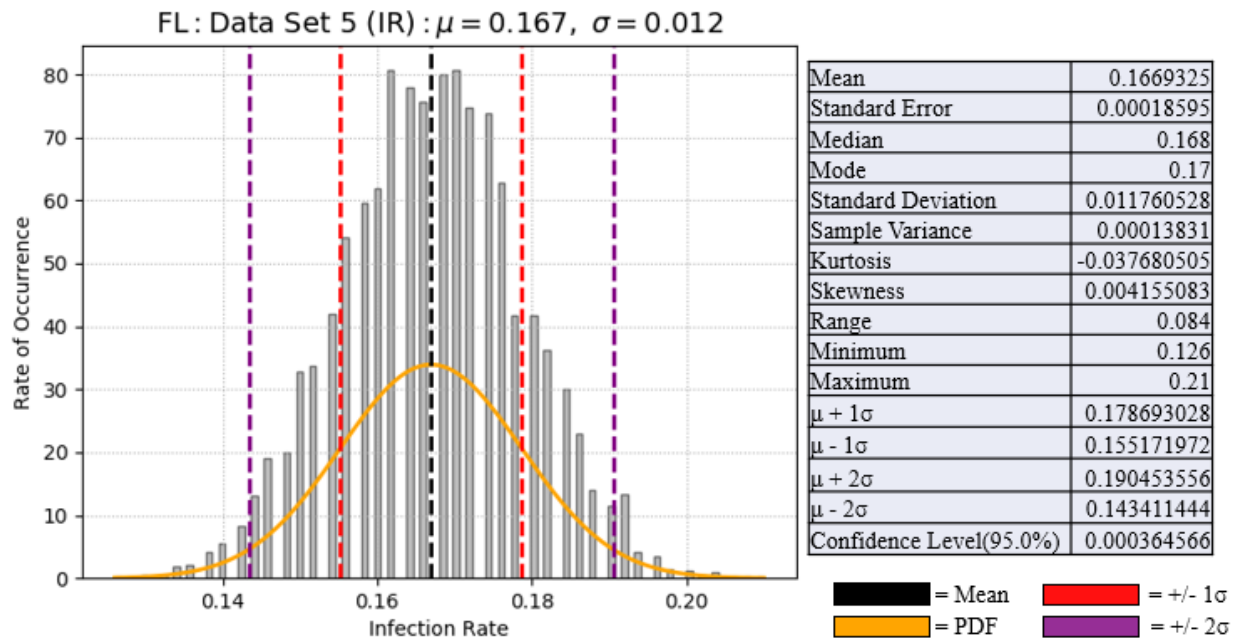


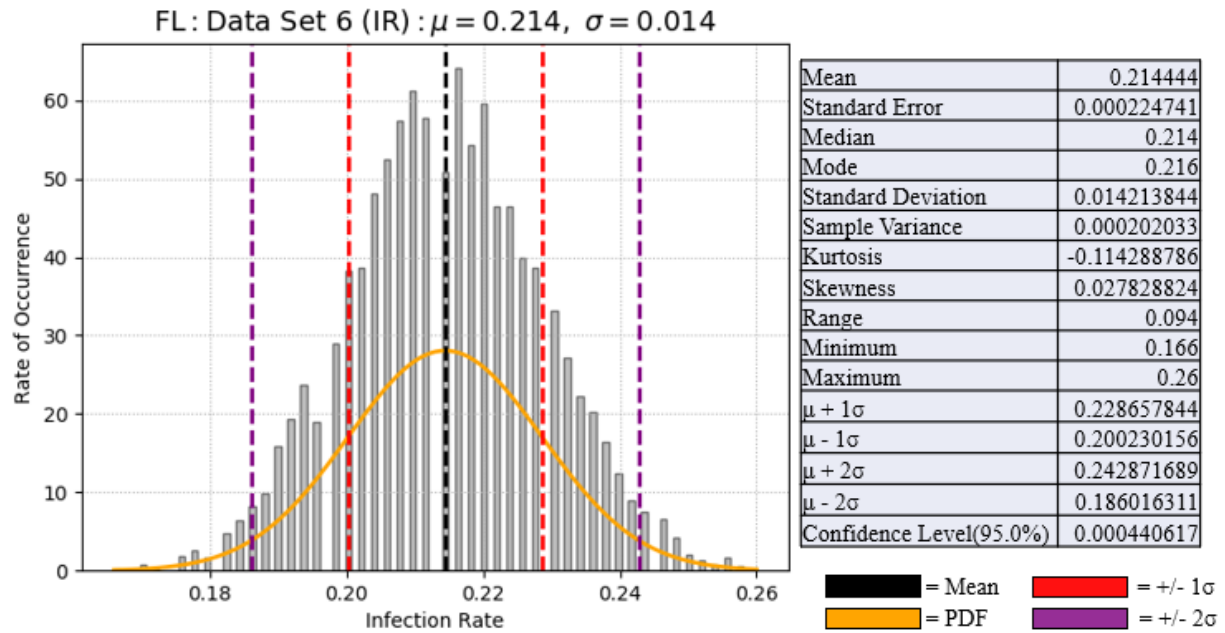
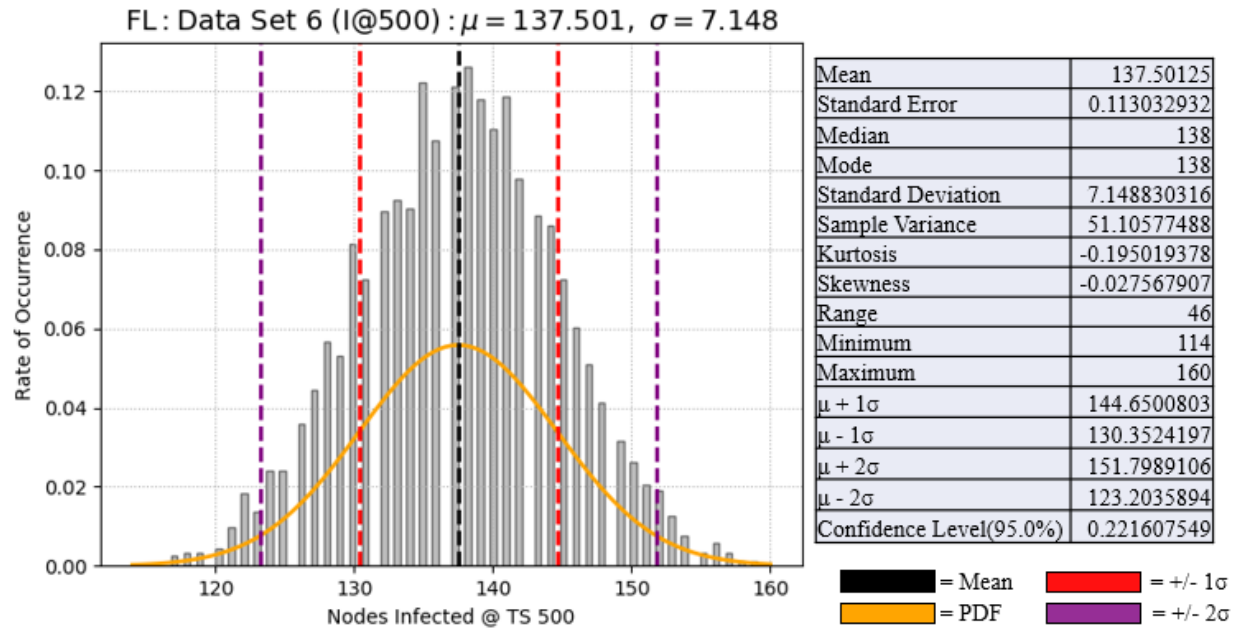


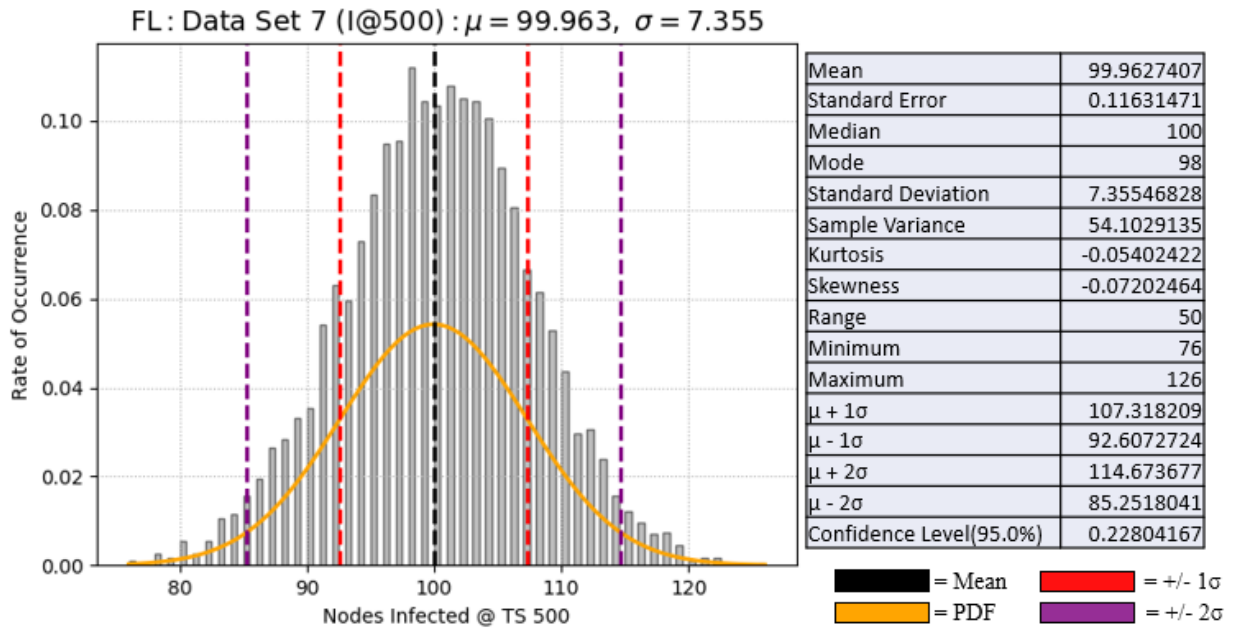
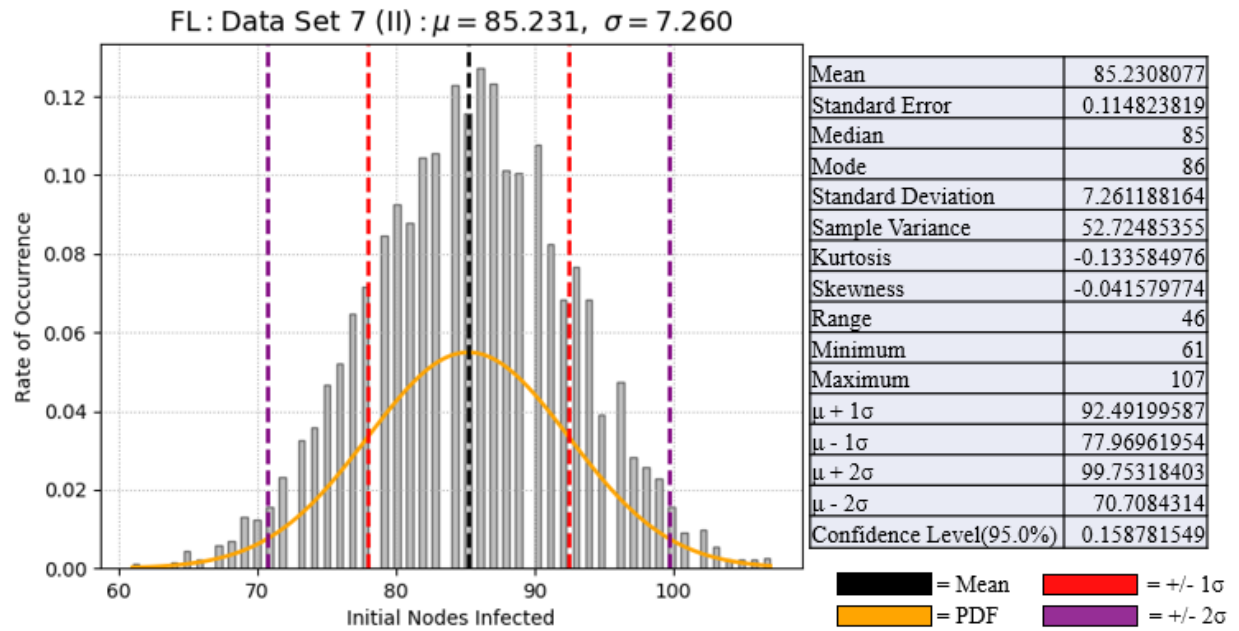


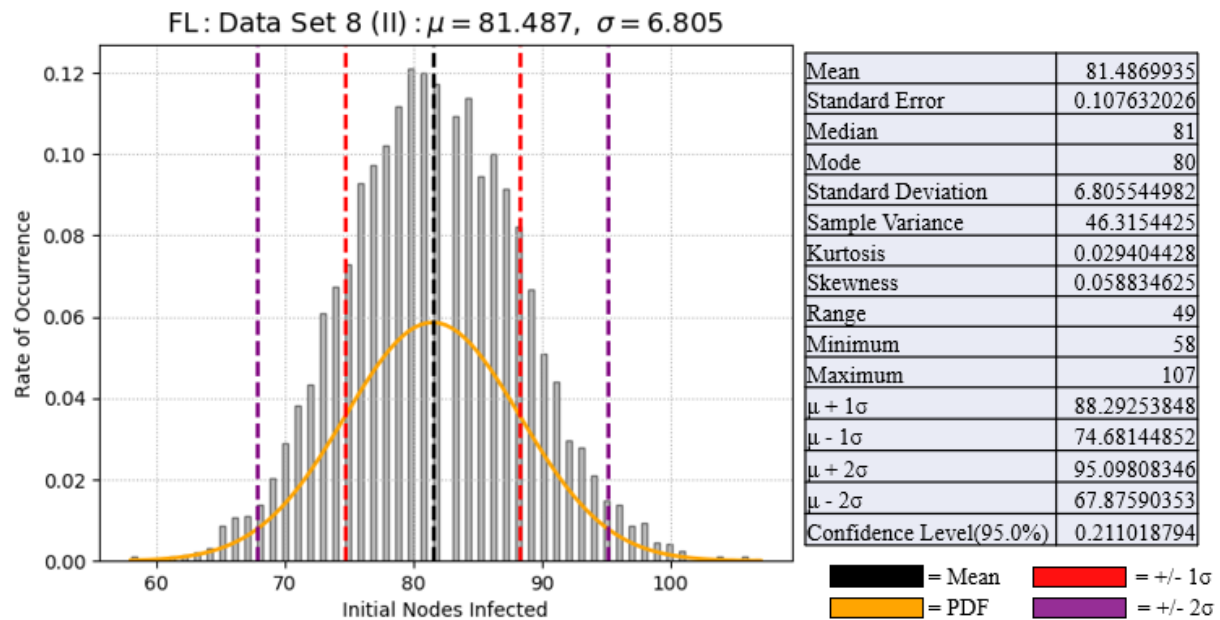
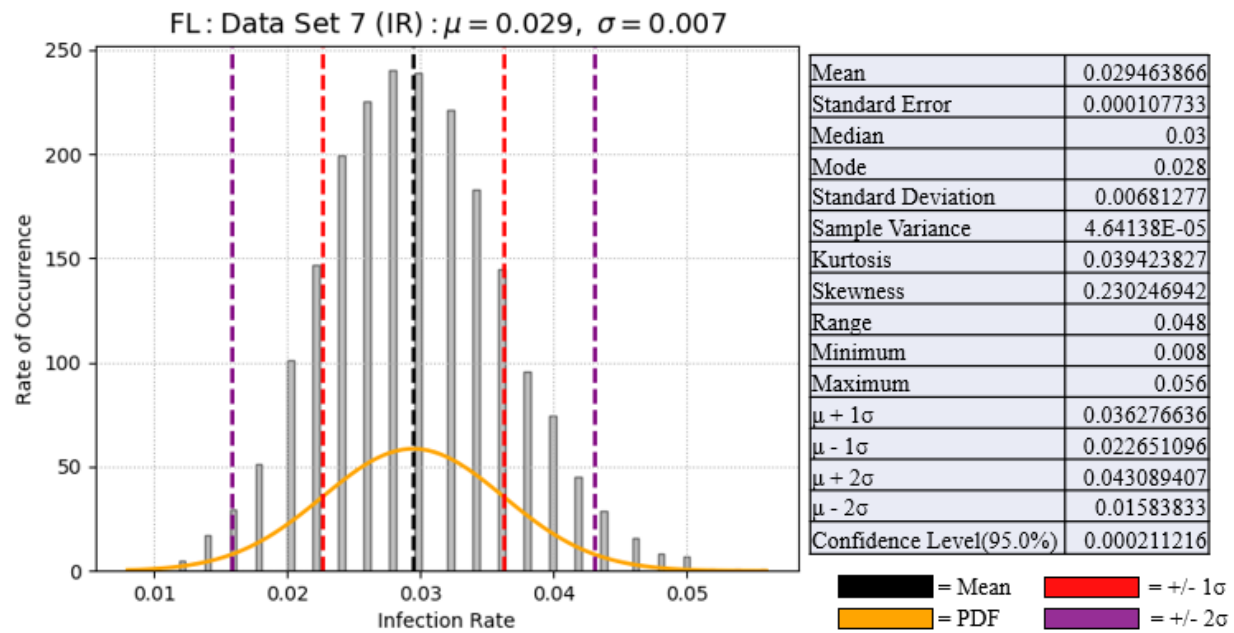


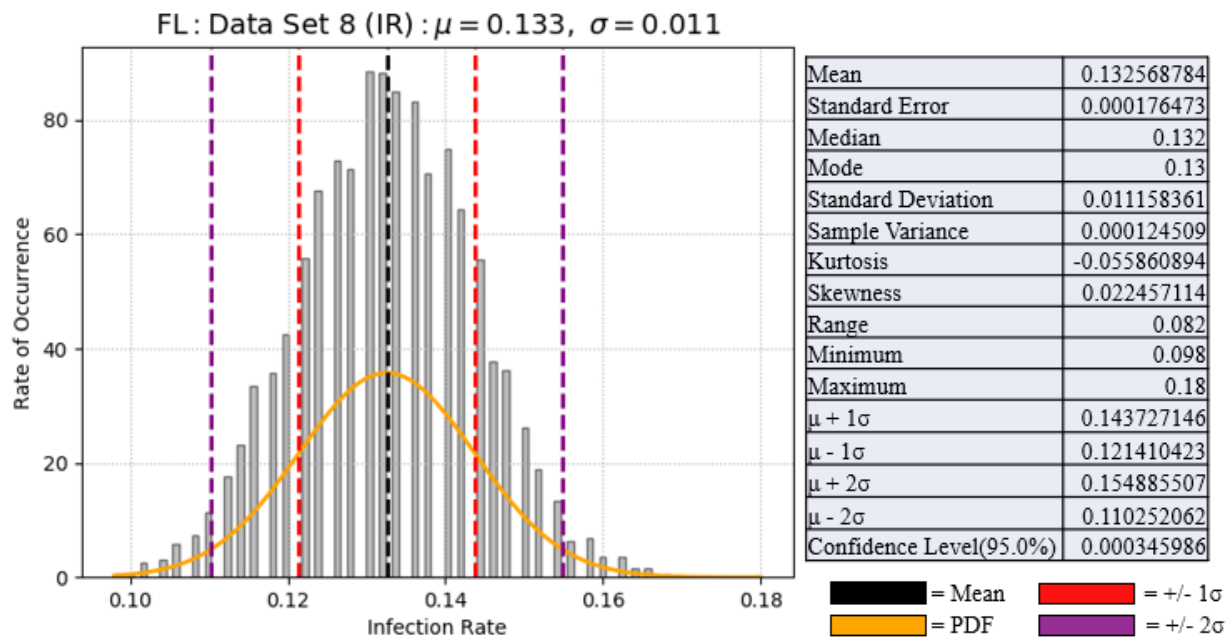
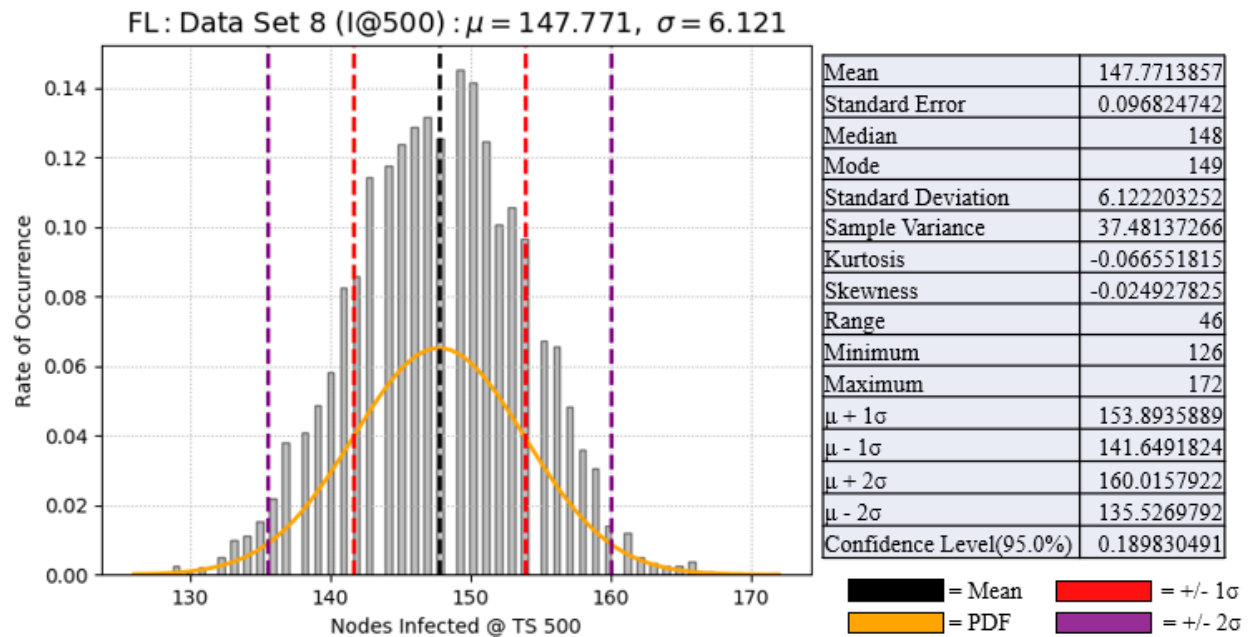


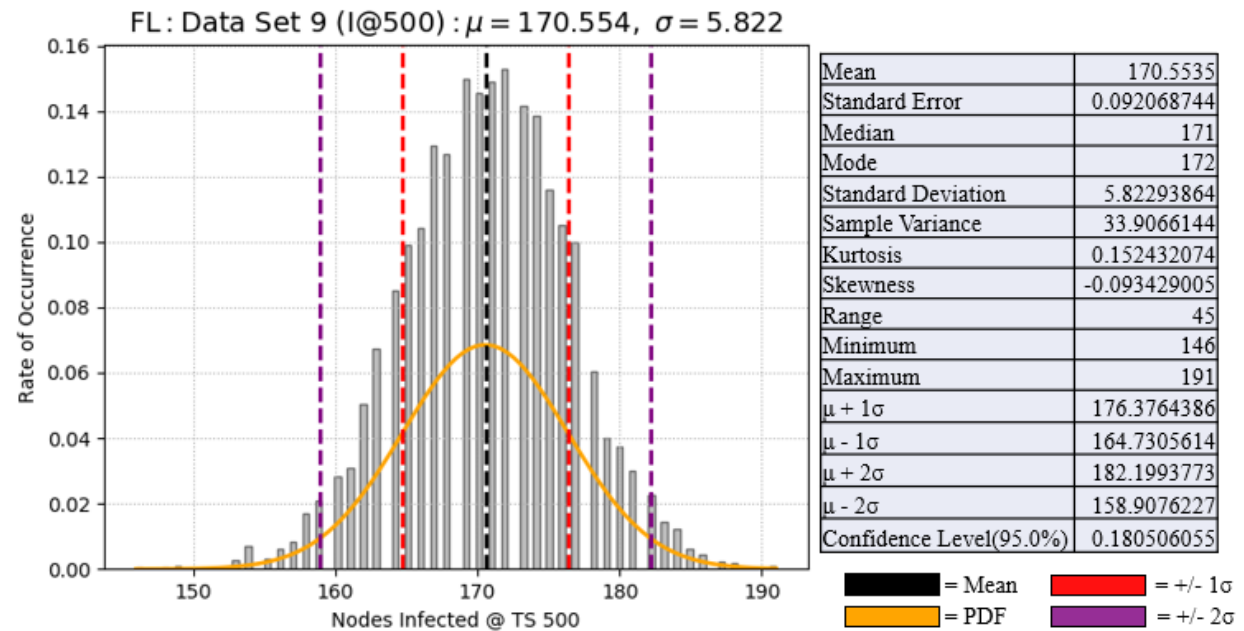
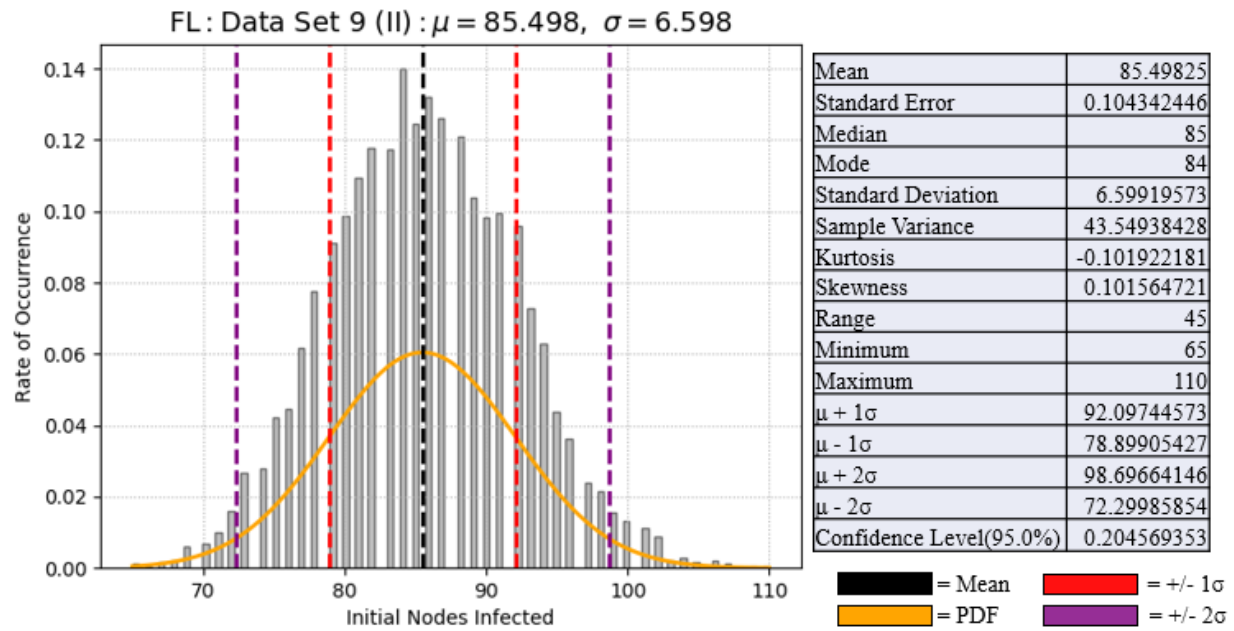


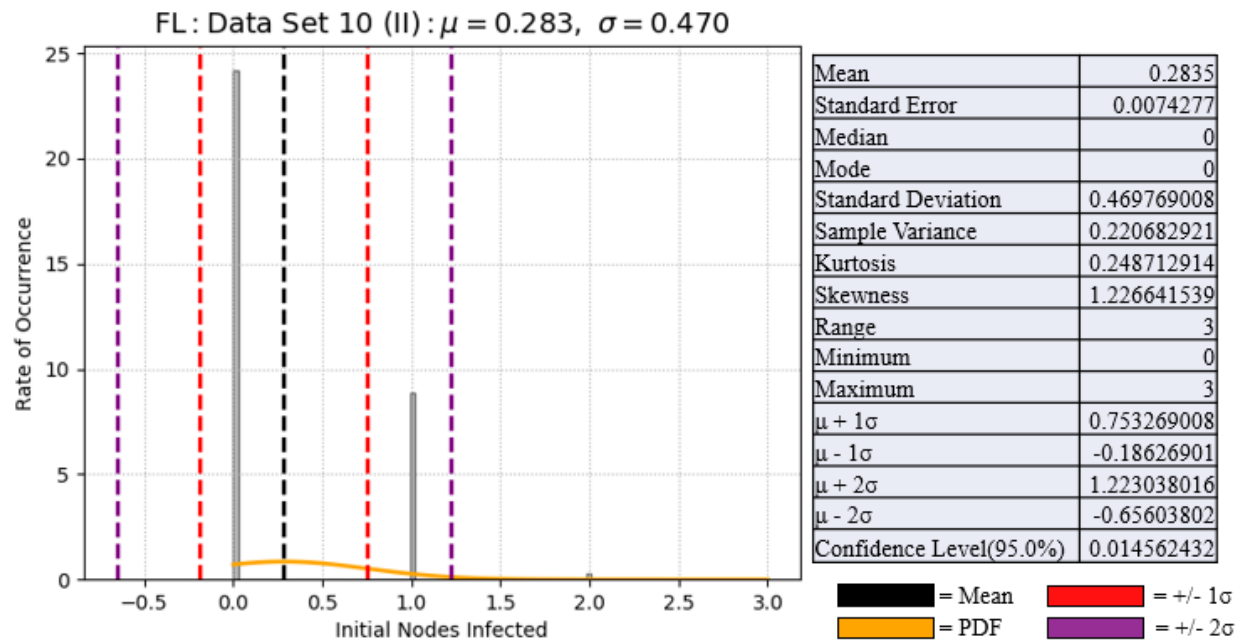
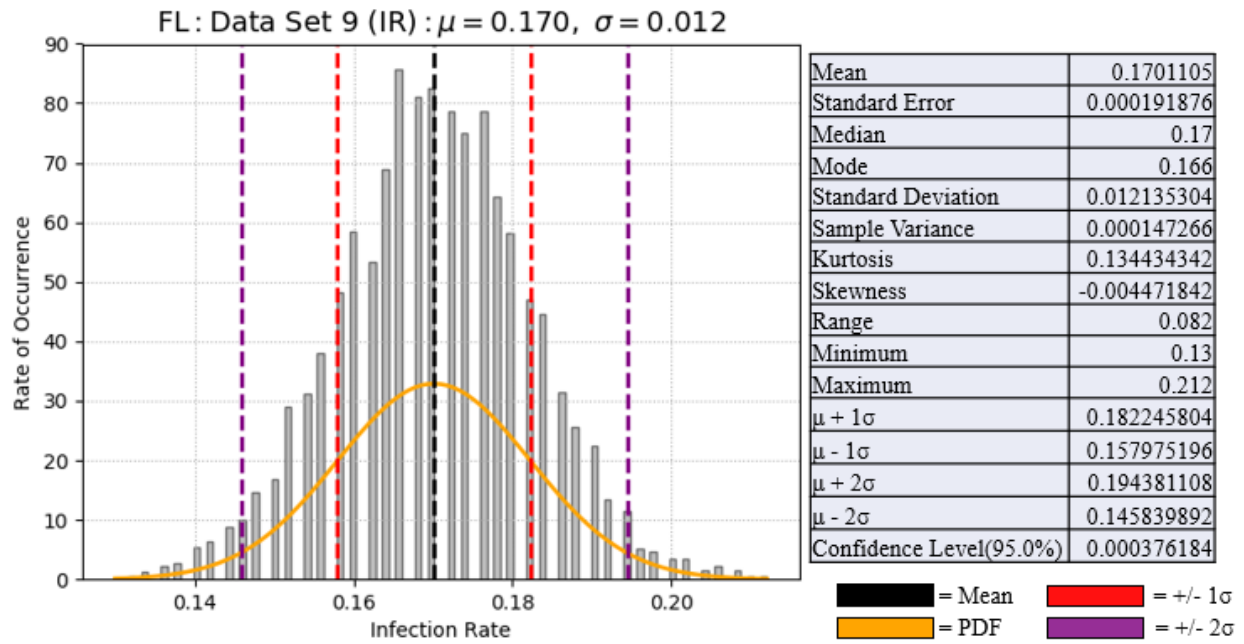


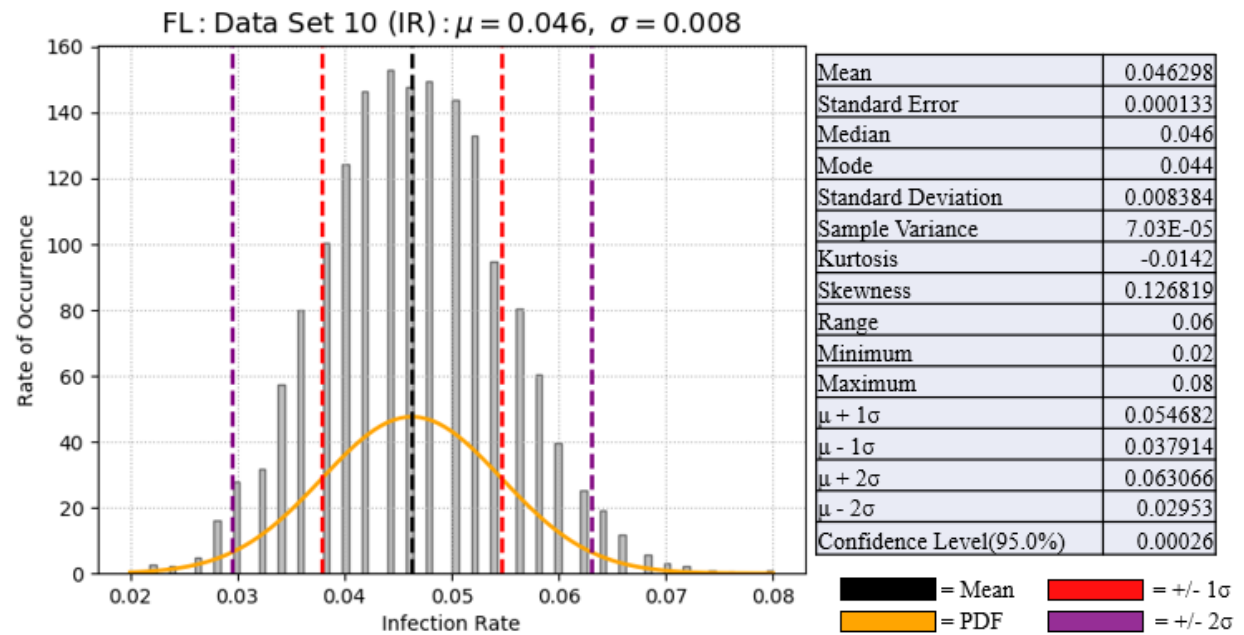
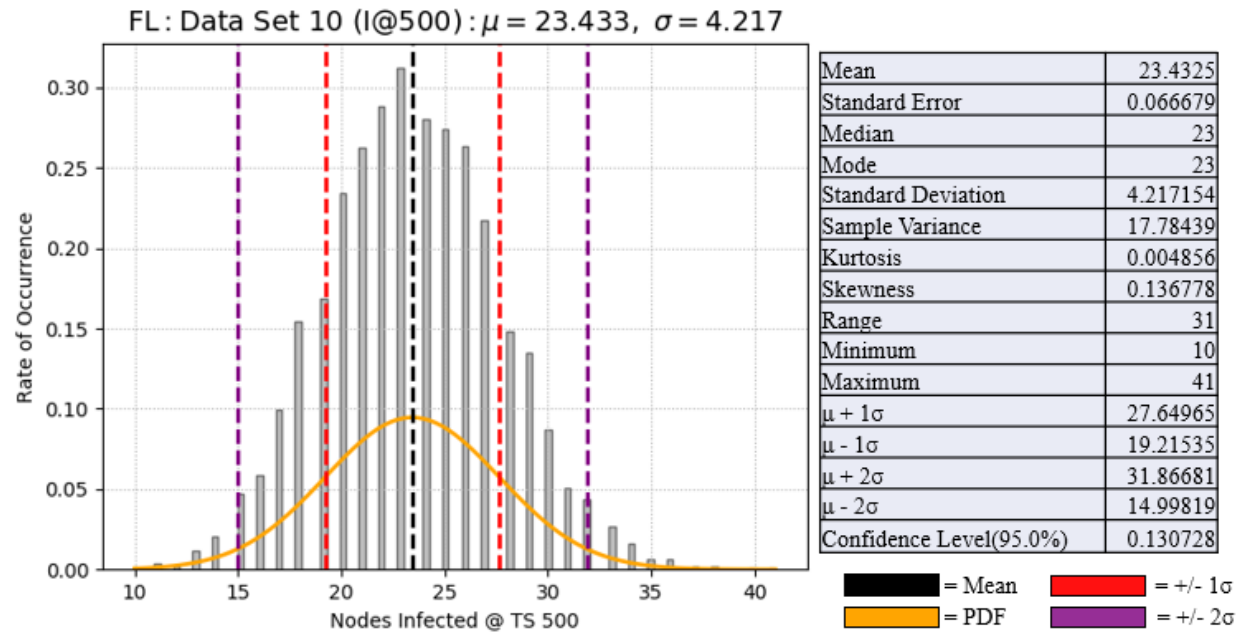




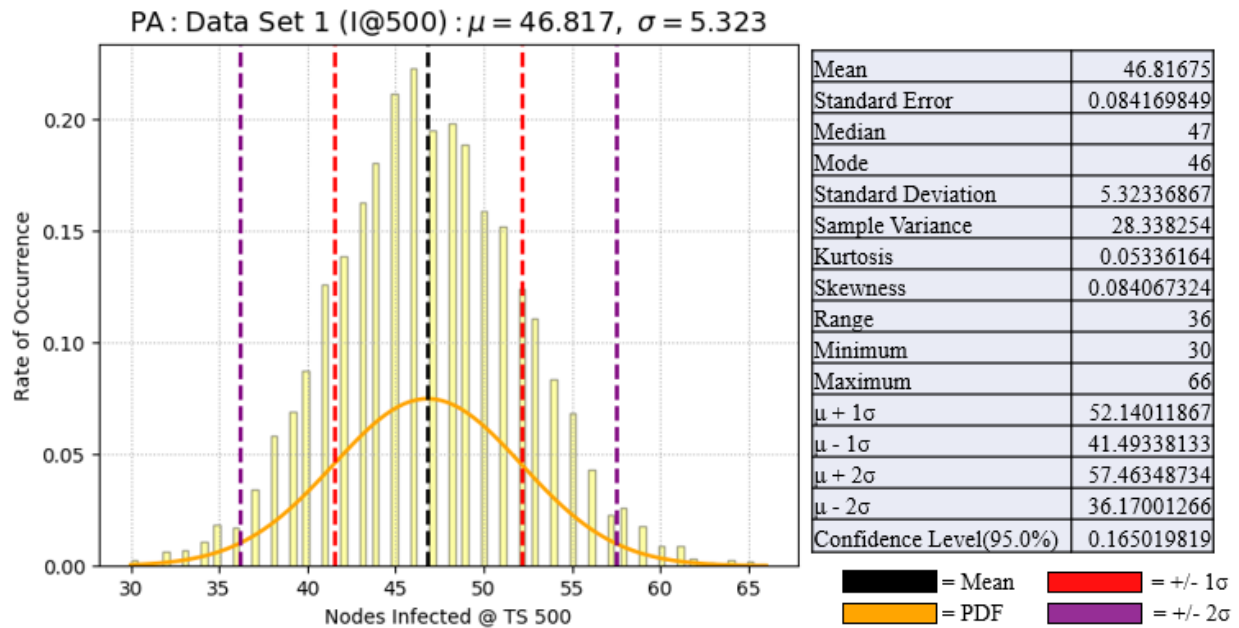
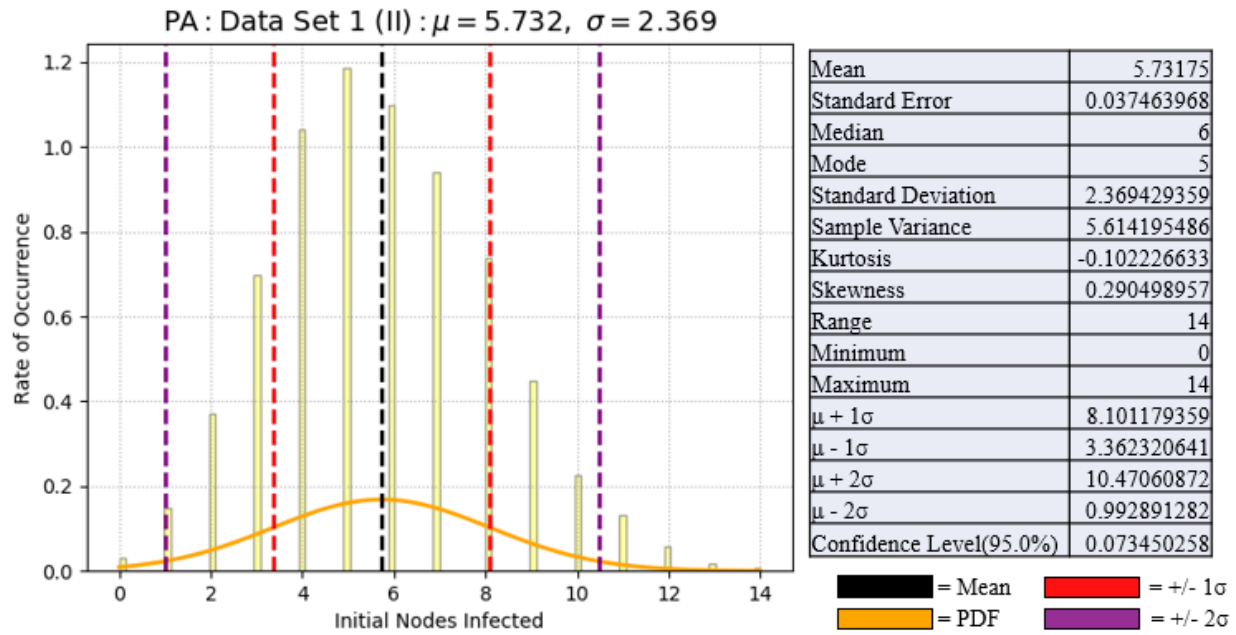


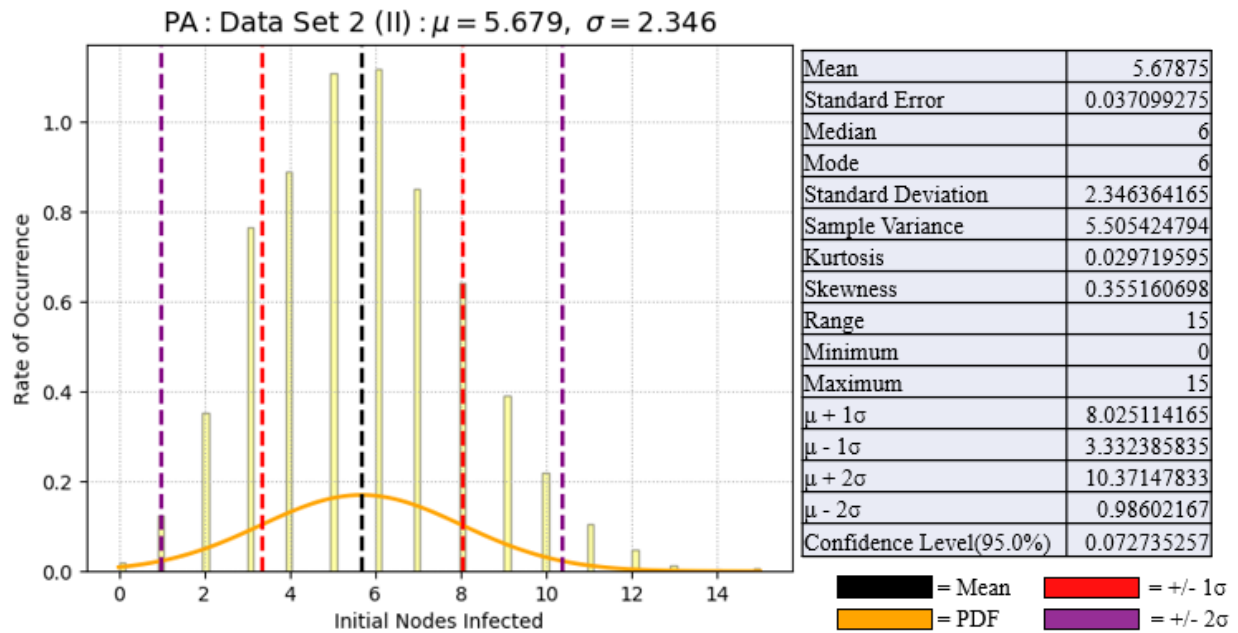
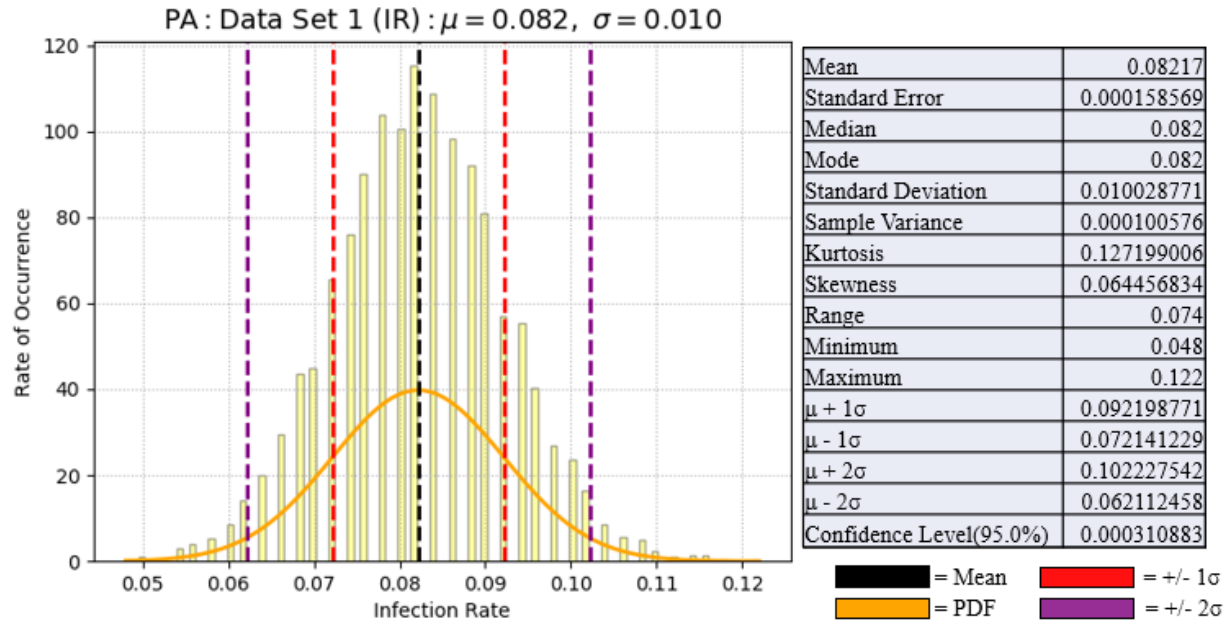


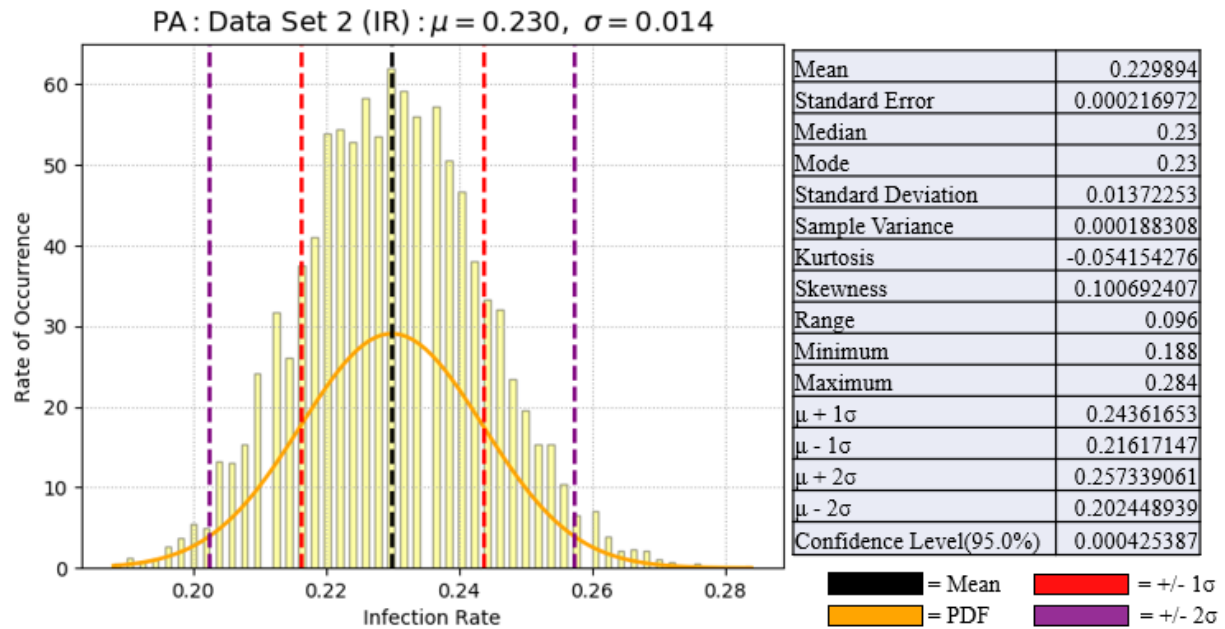
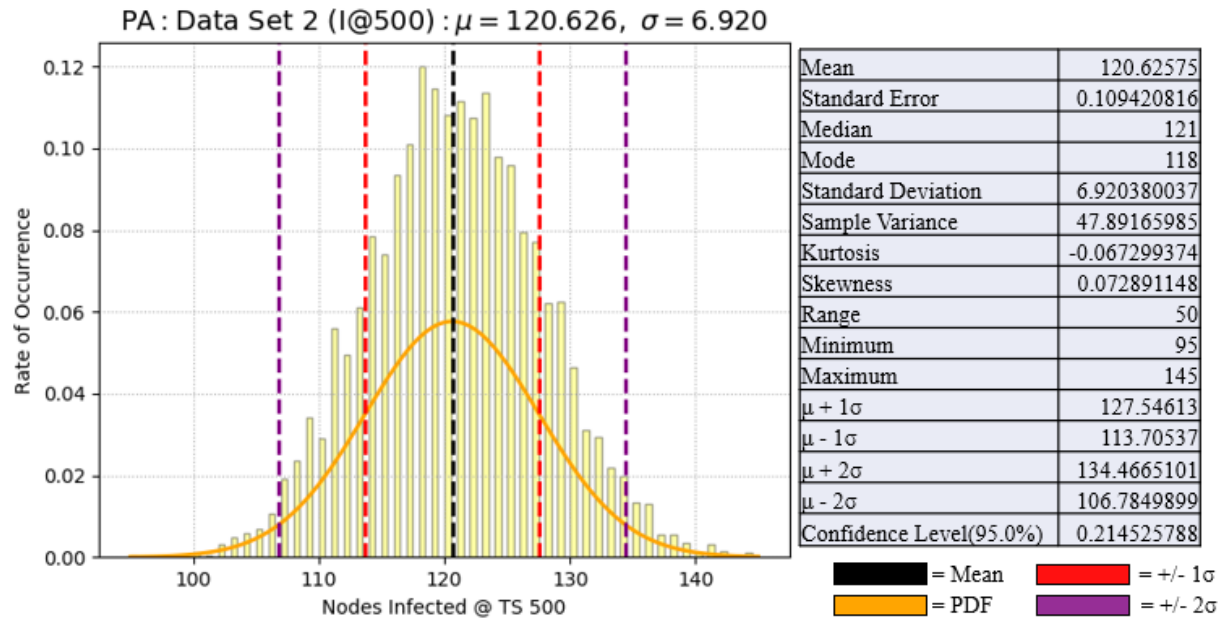


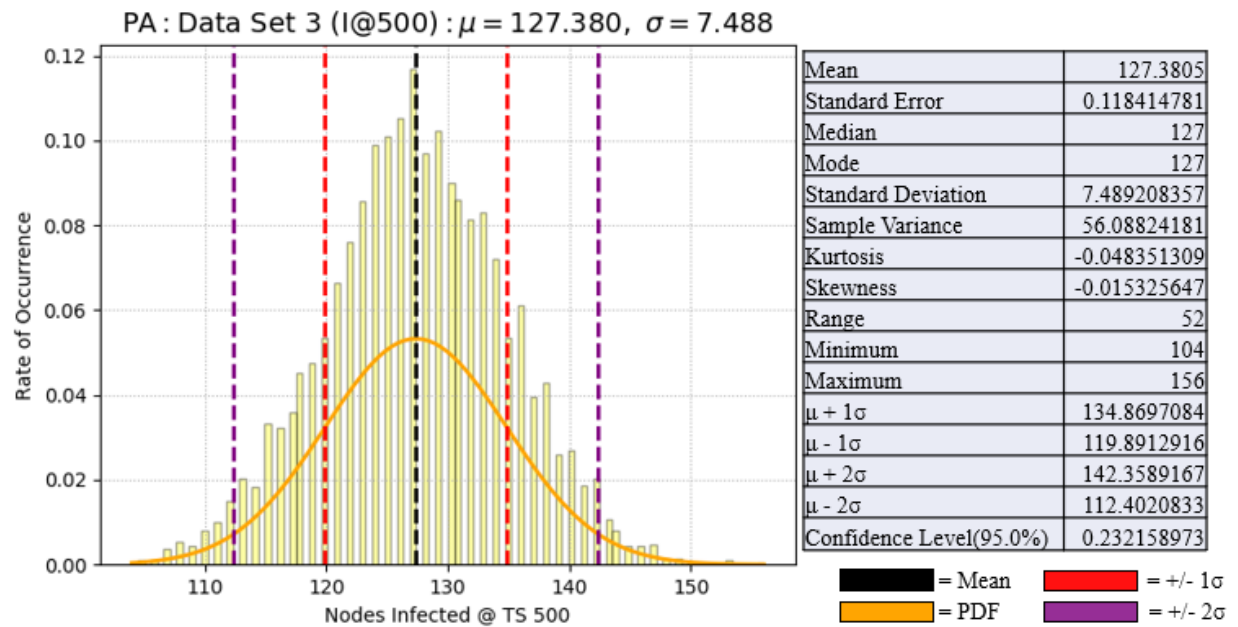
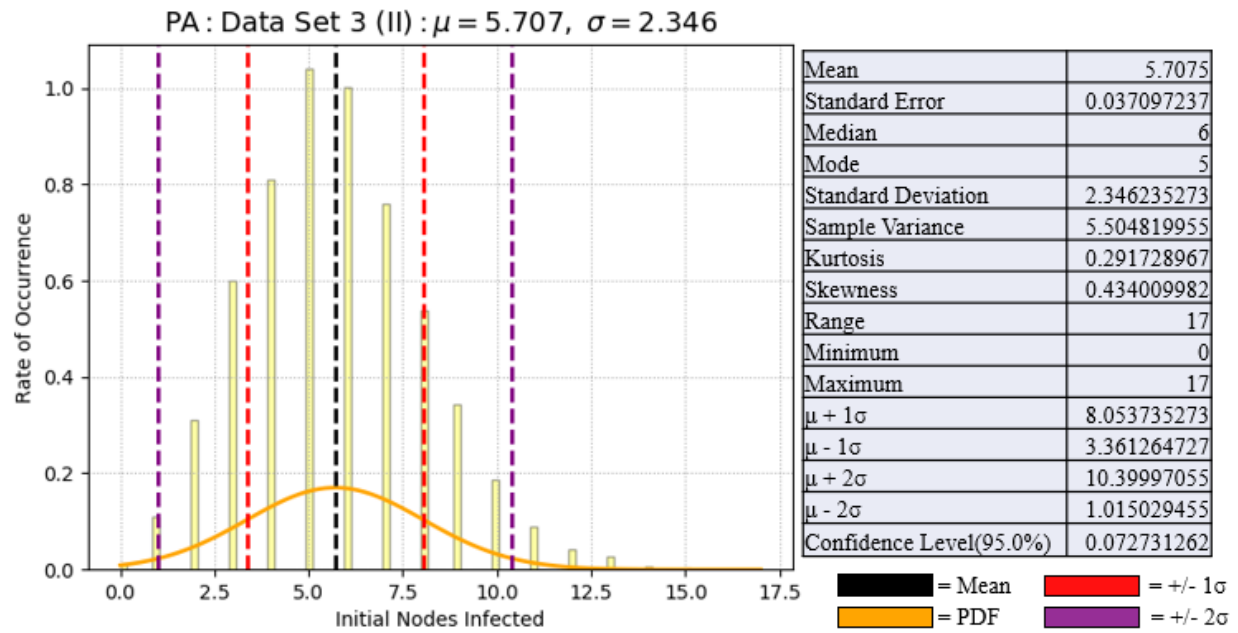


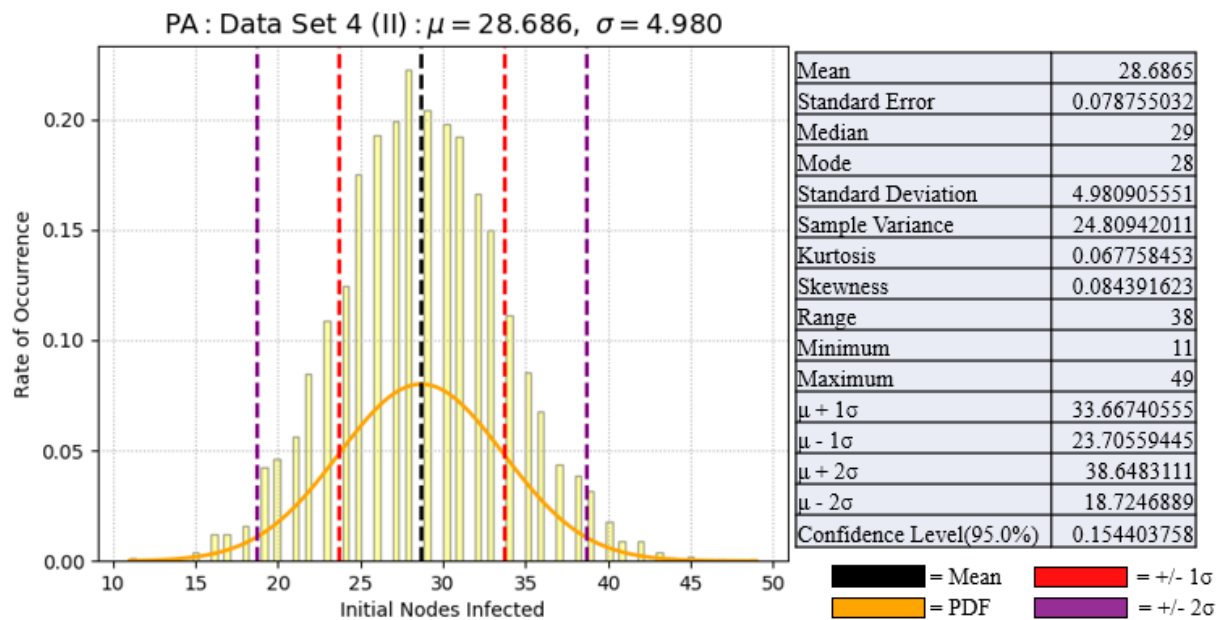
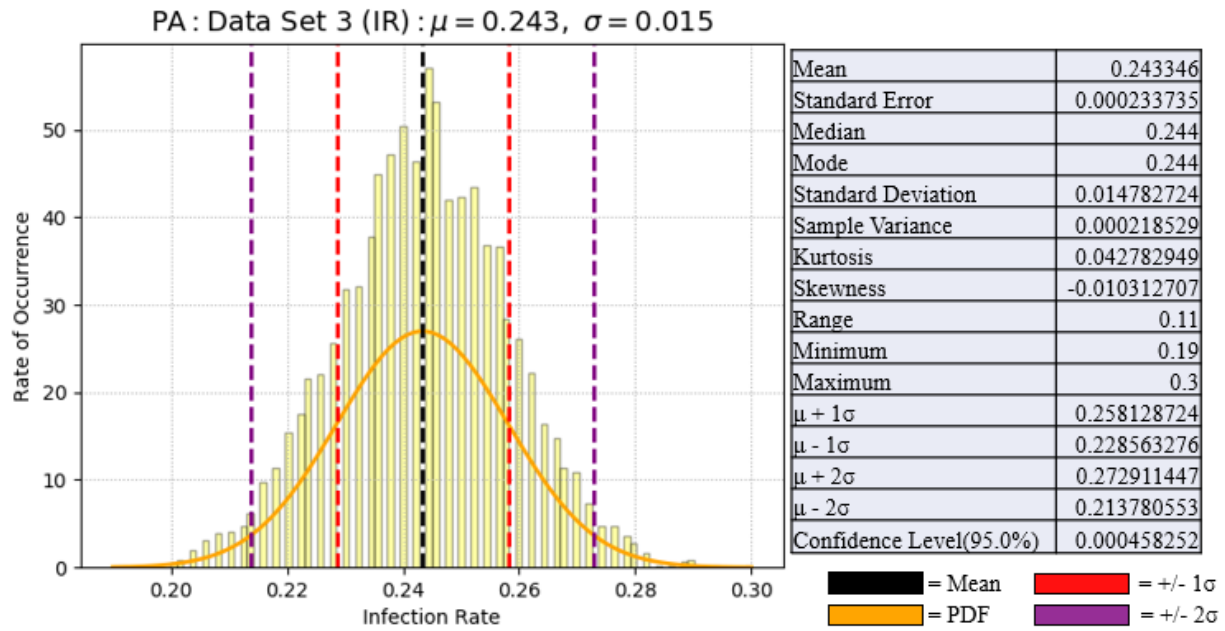
APPENDIX E. PENNSYLVANIA NETWORK DATA GRAPHS AND STATISTICS

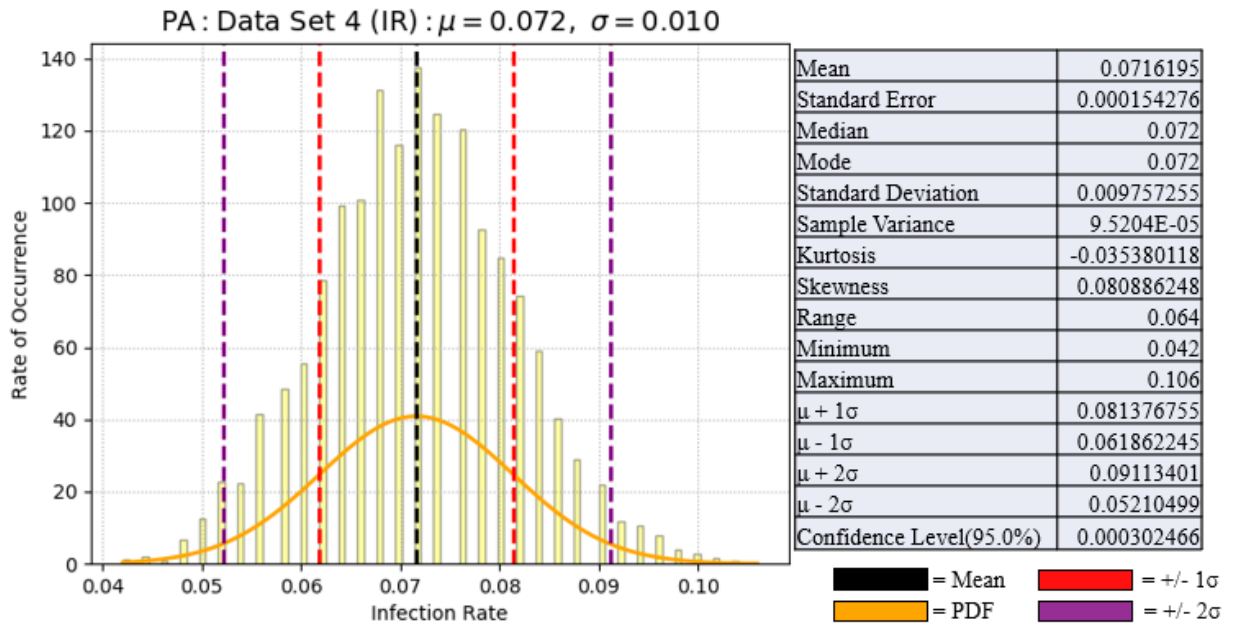
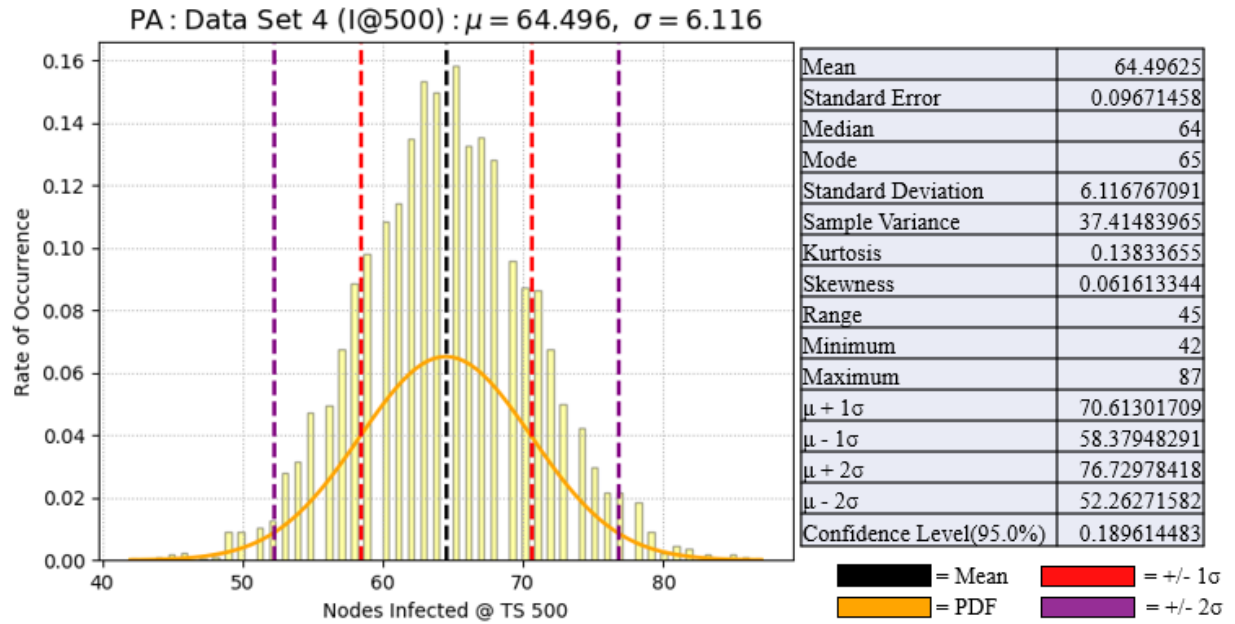


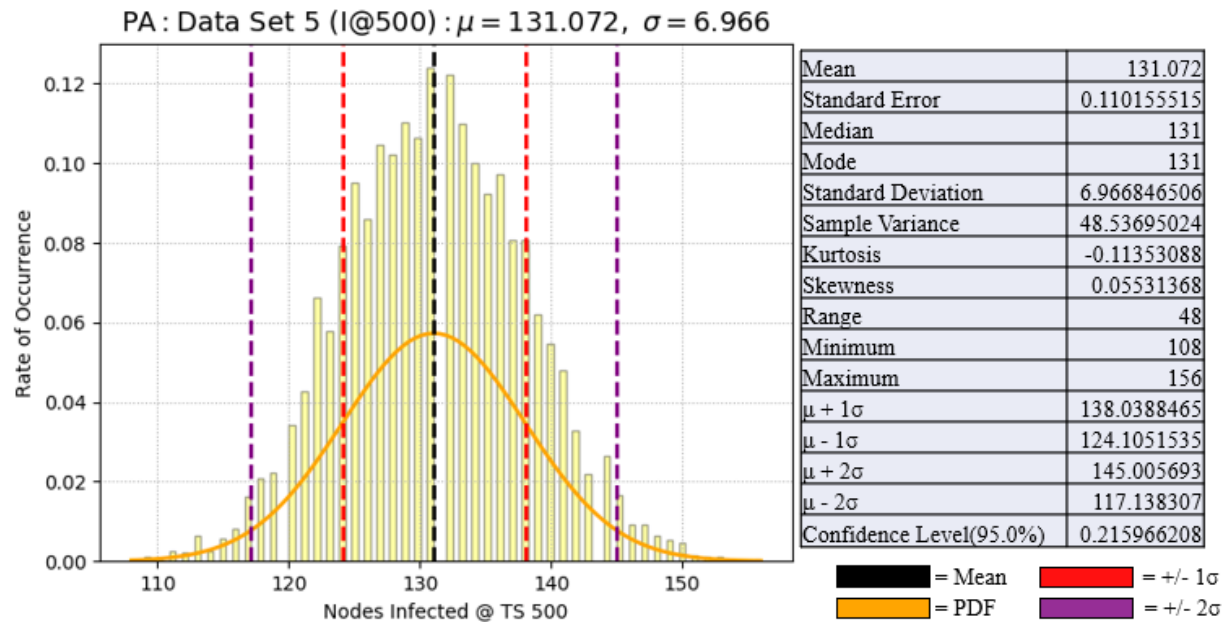
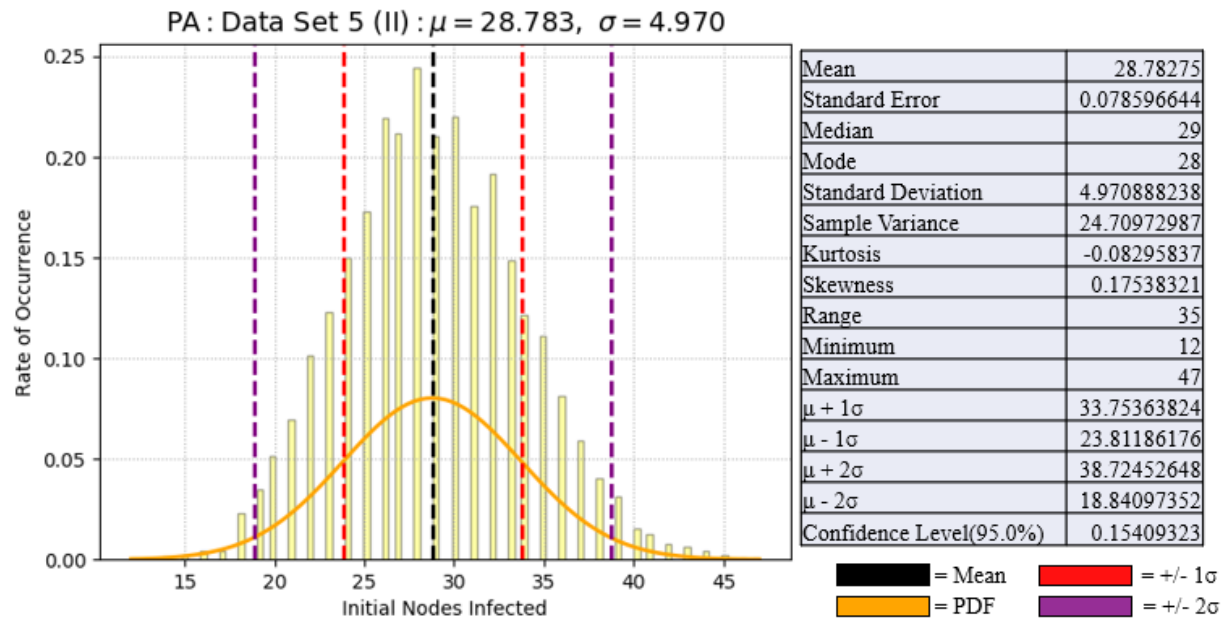


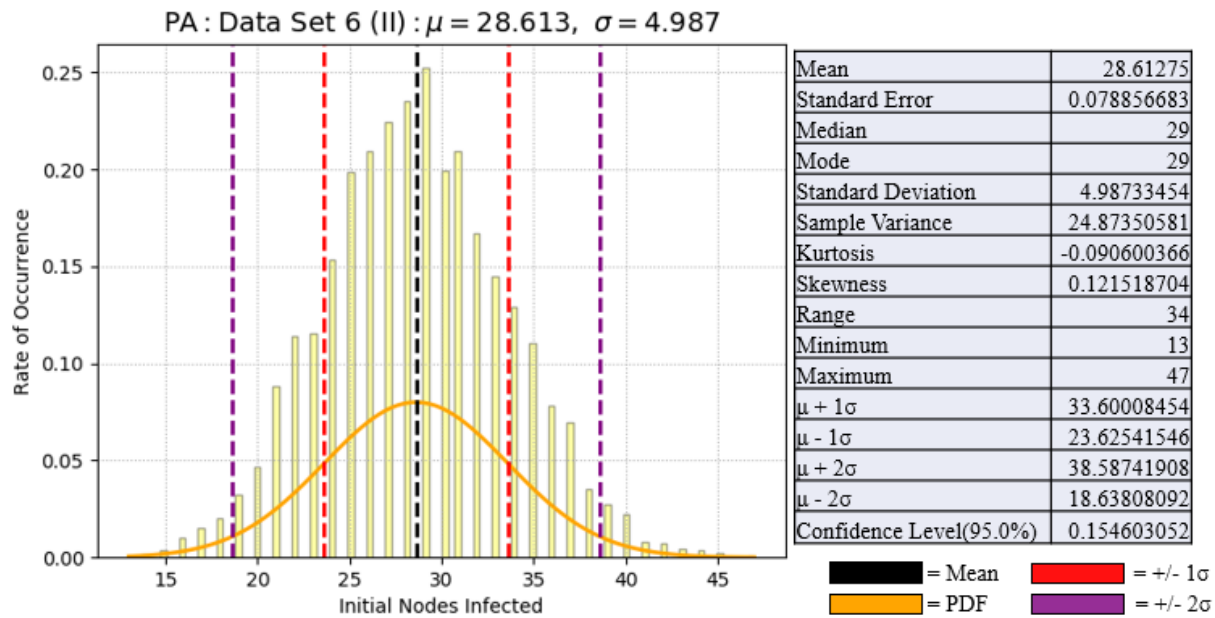
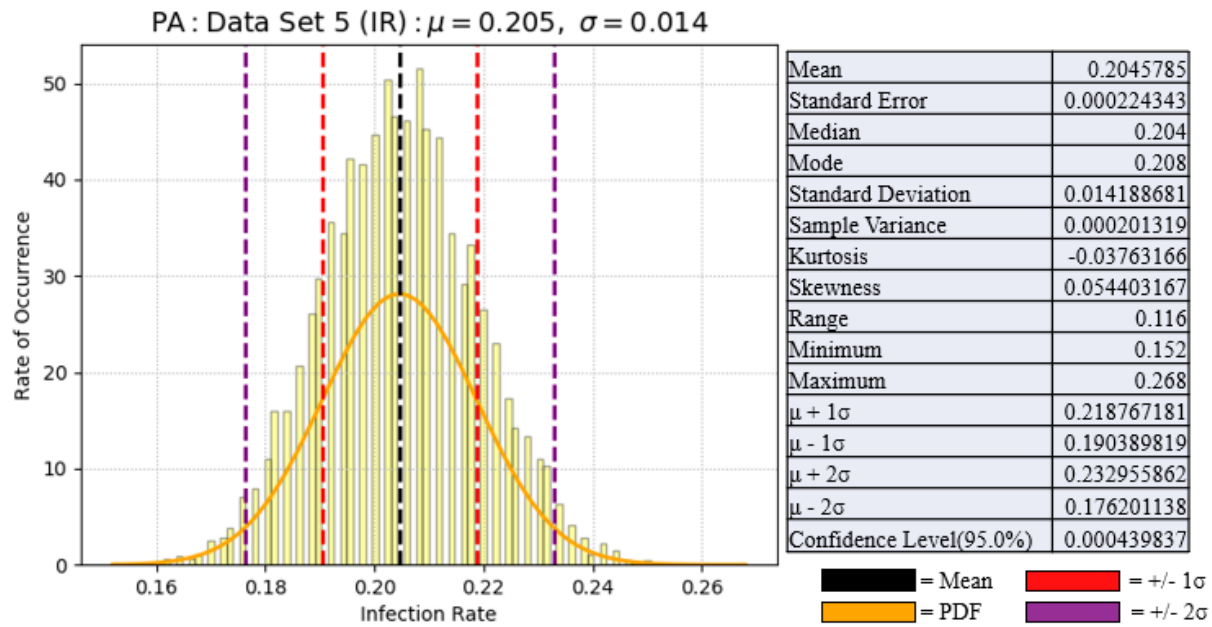


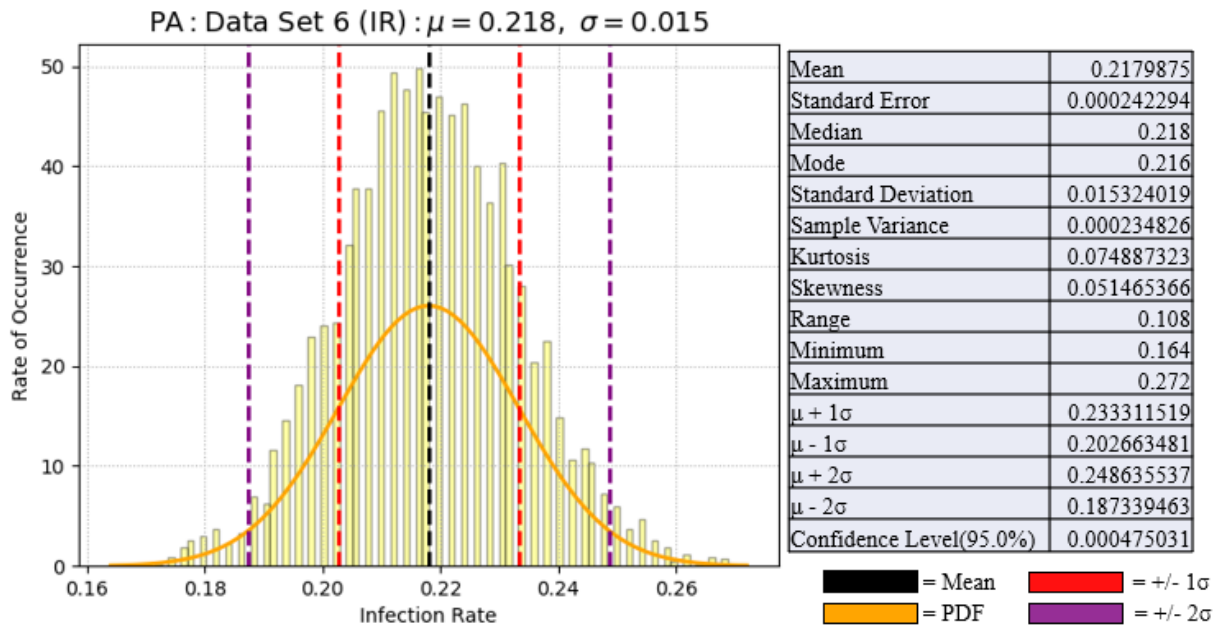
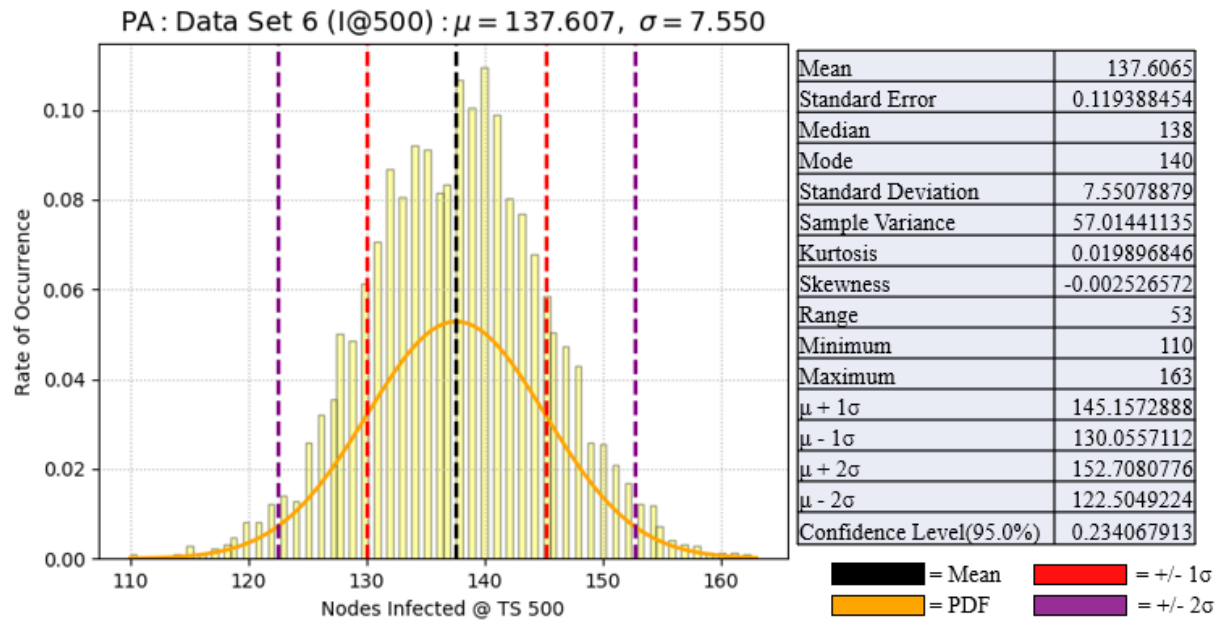


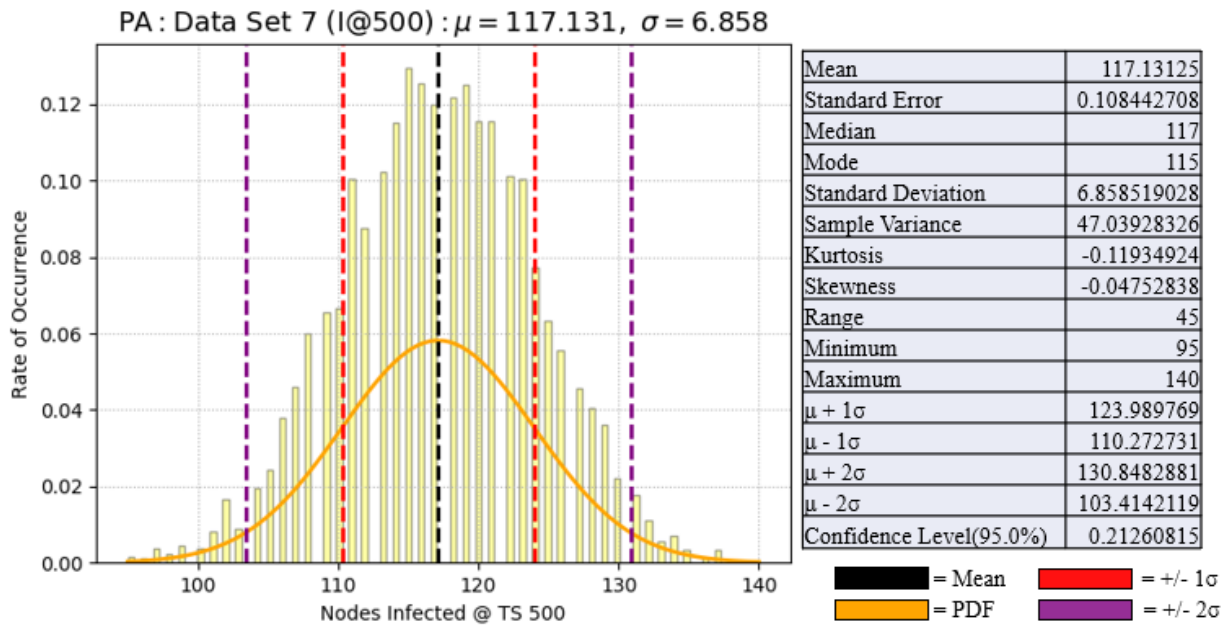
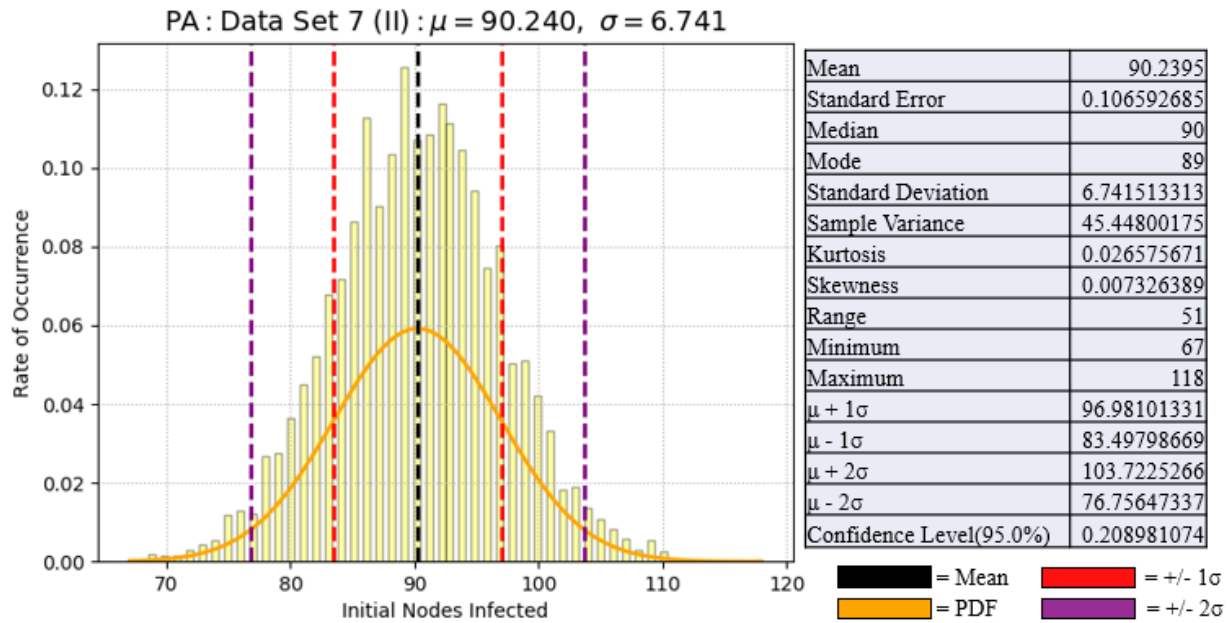


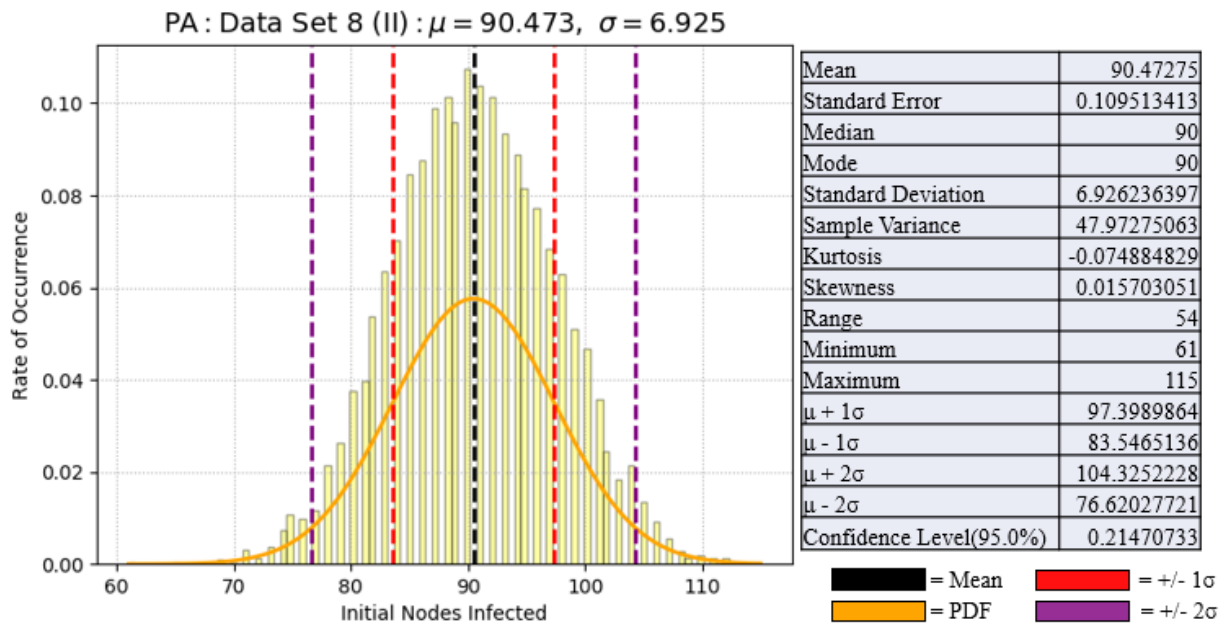
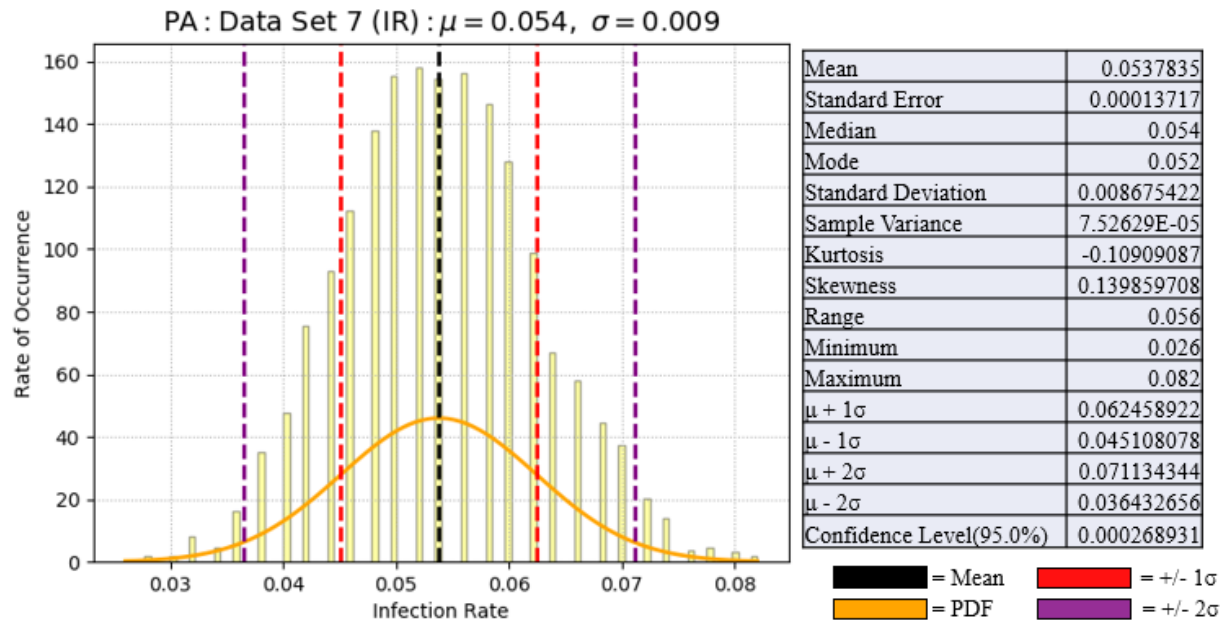


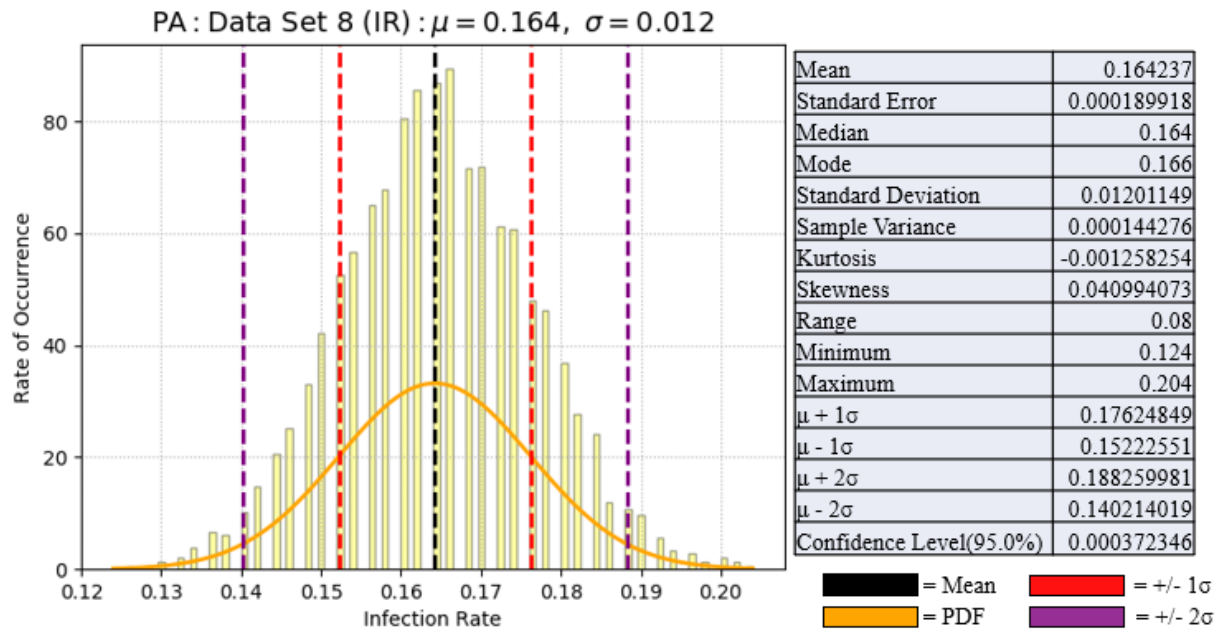
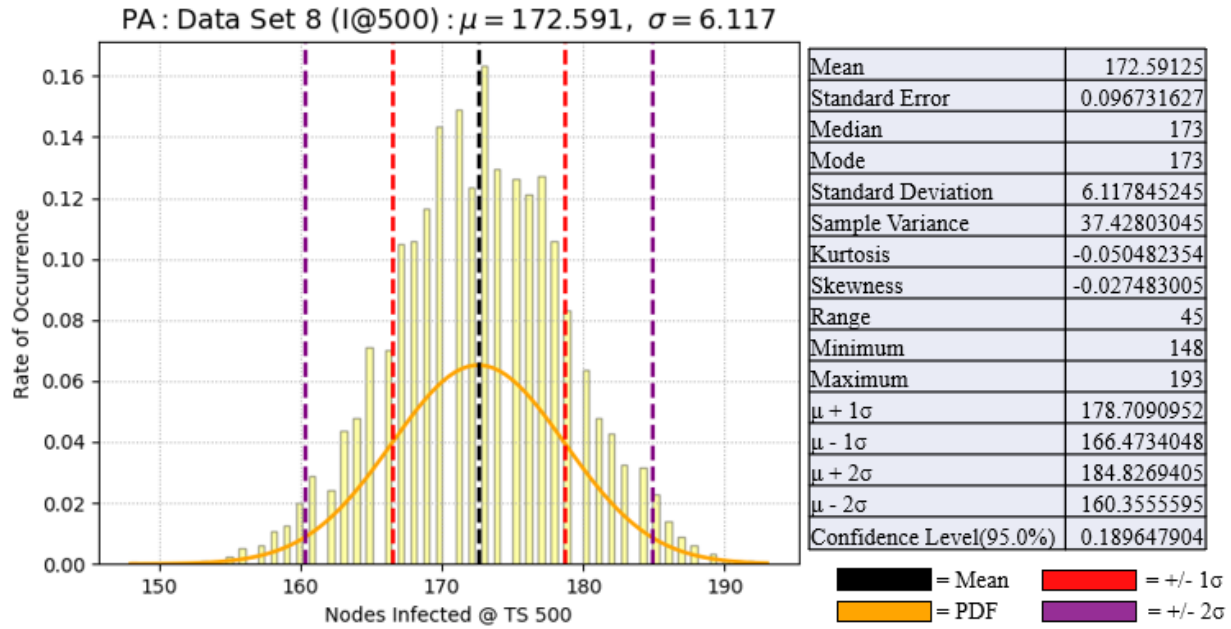


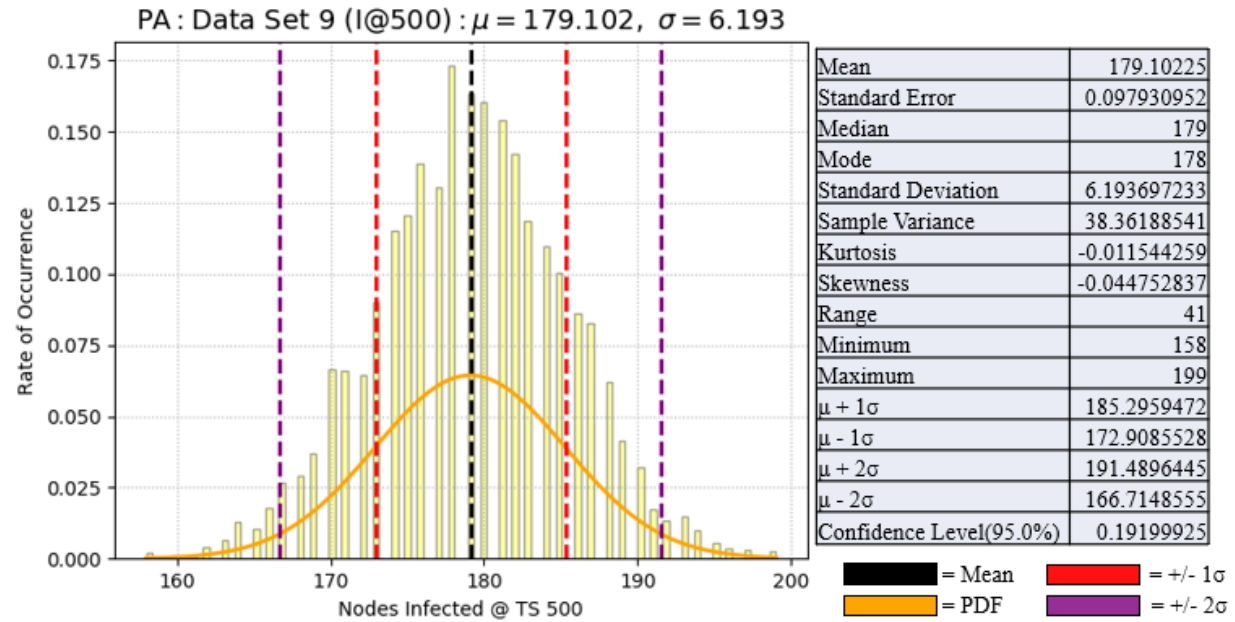
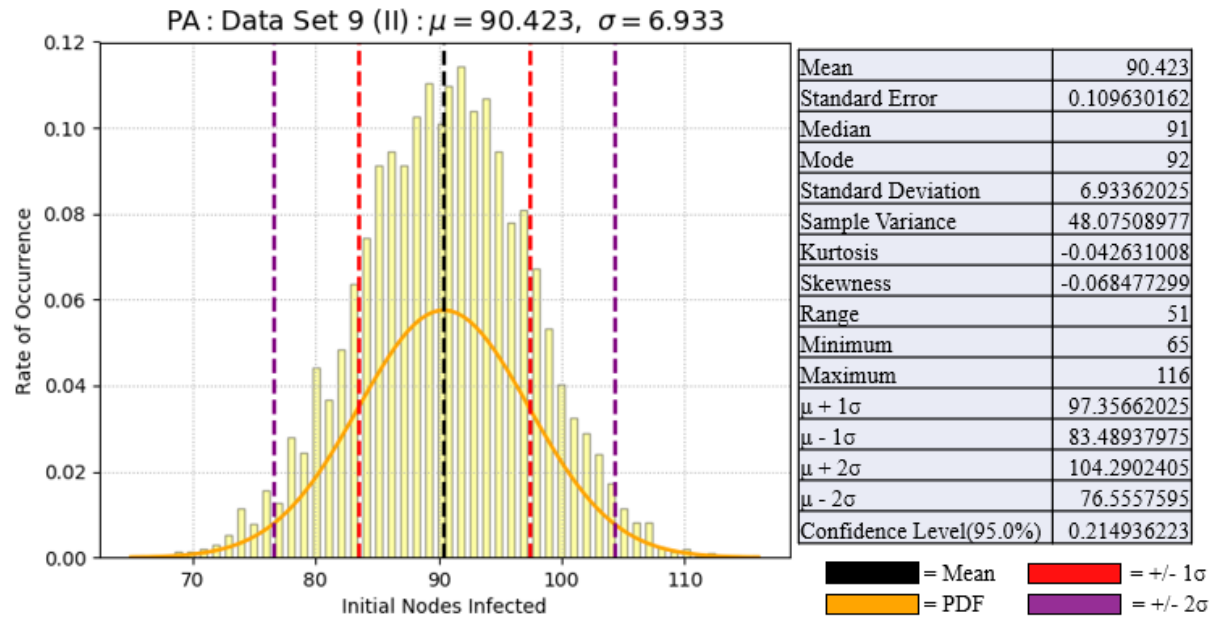


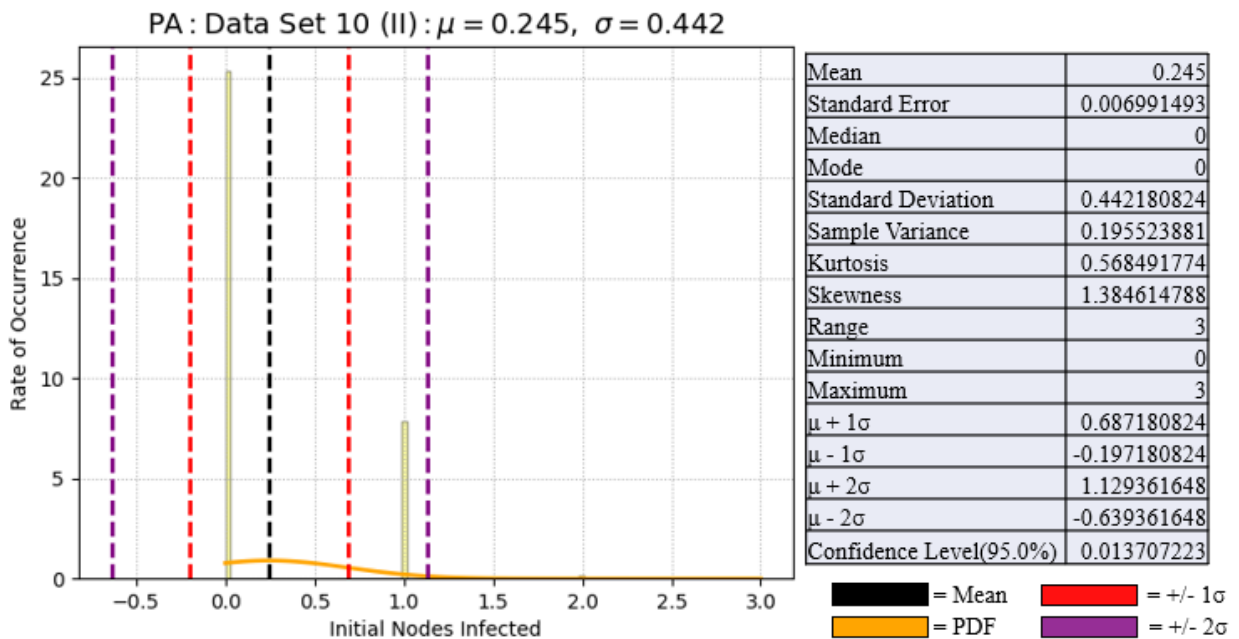
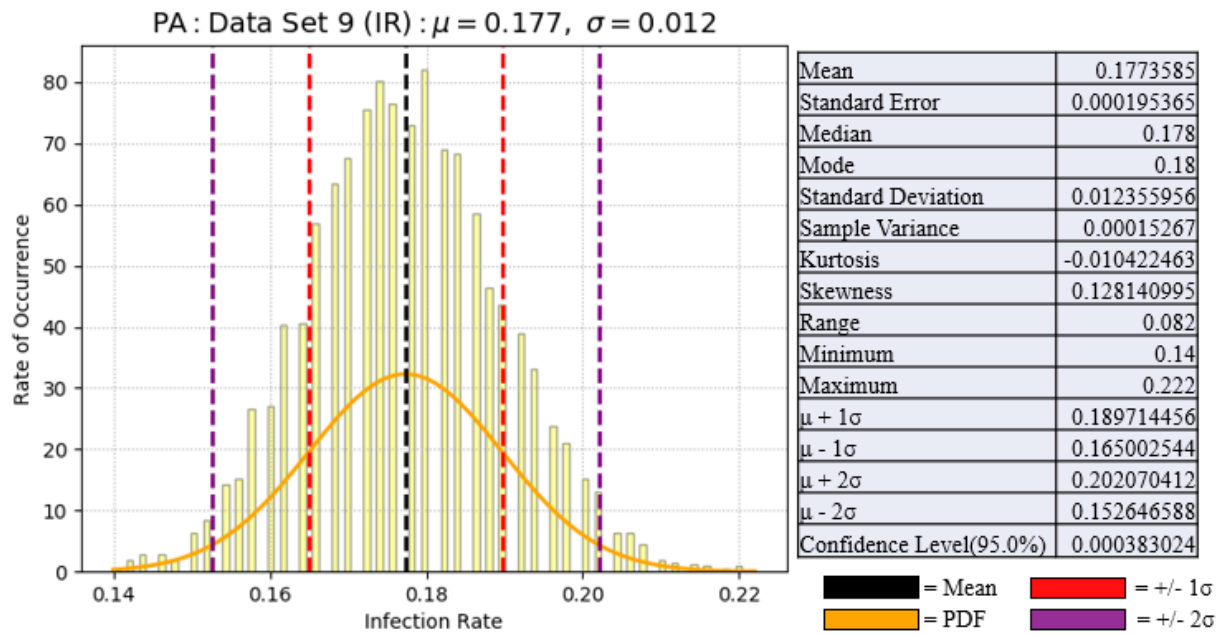


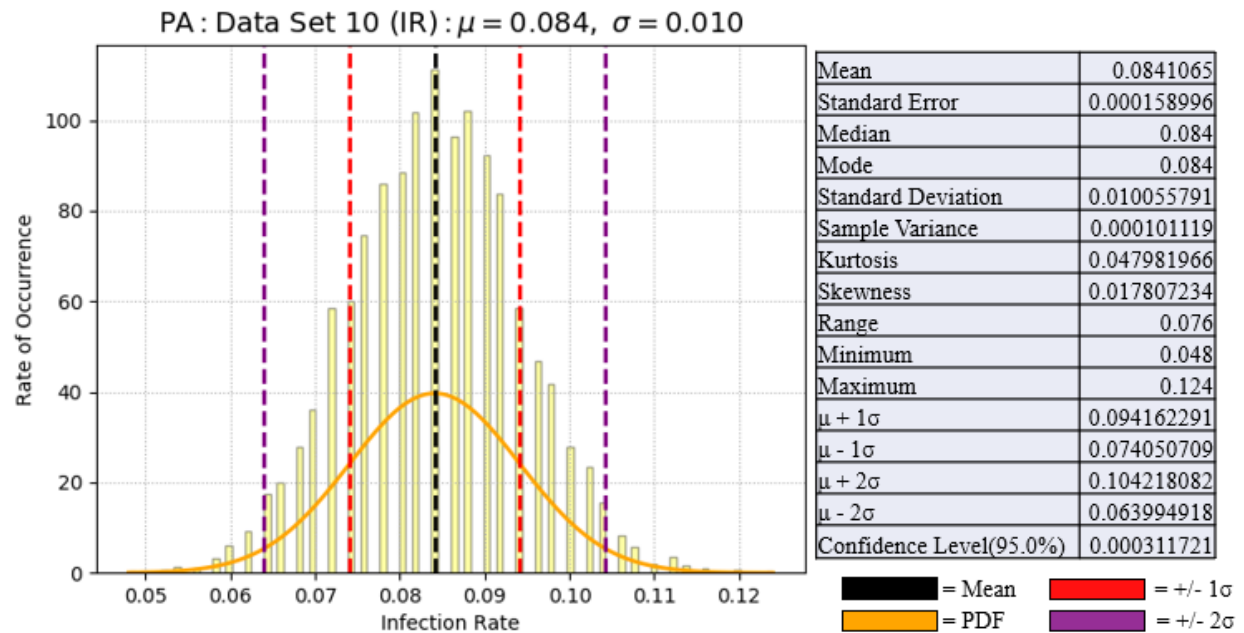
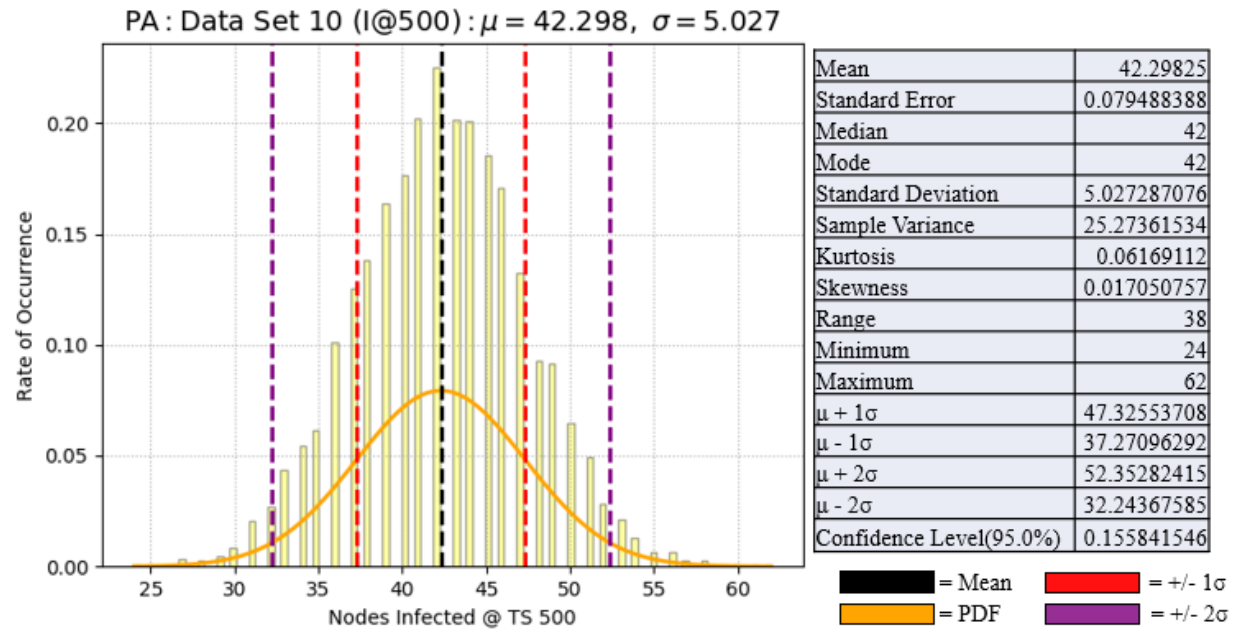




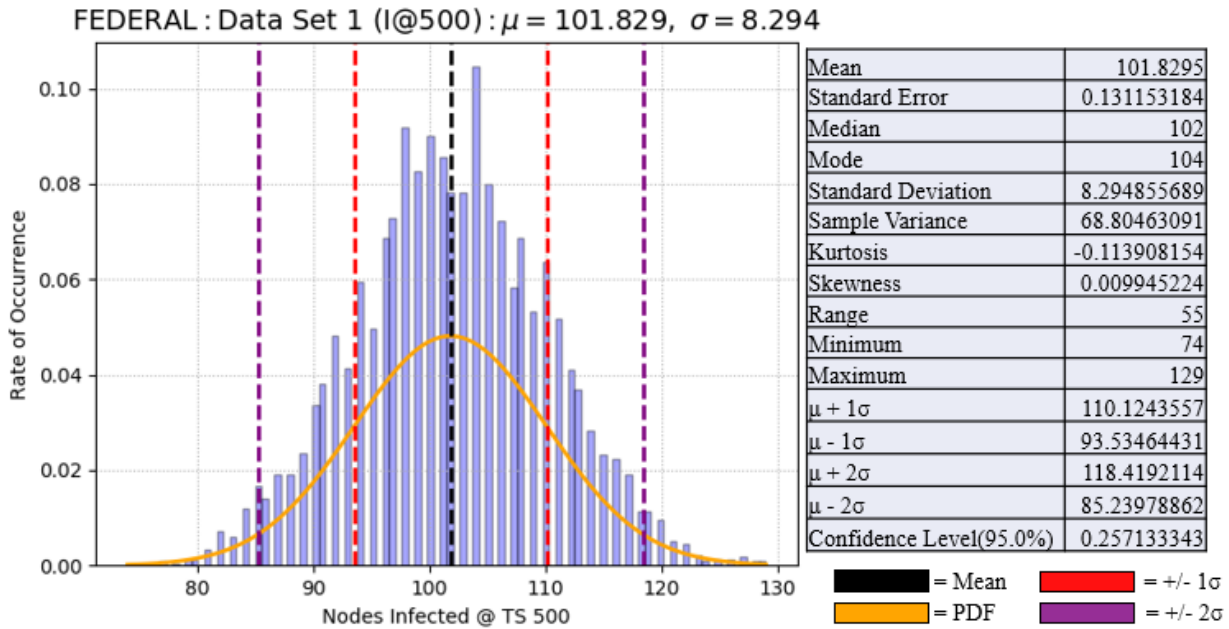
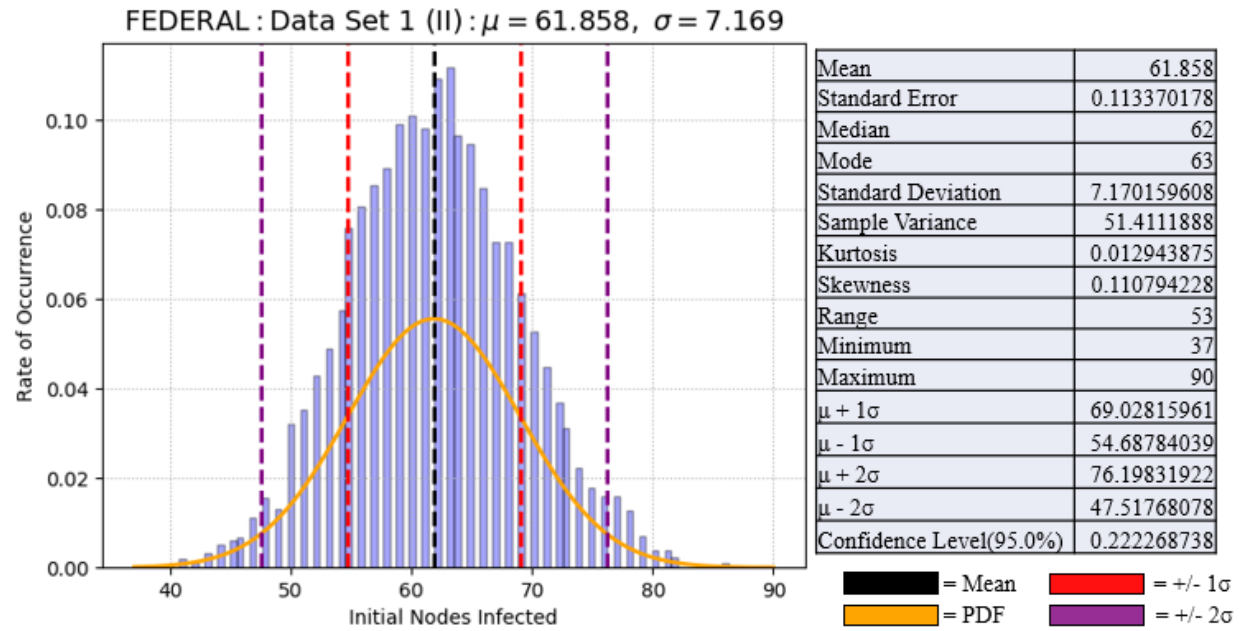


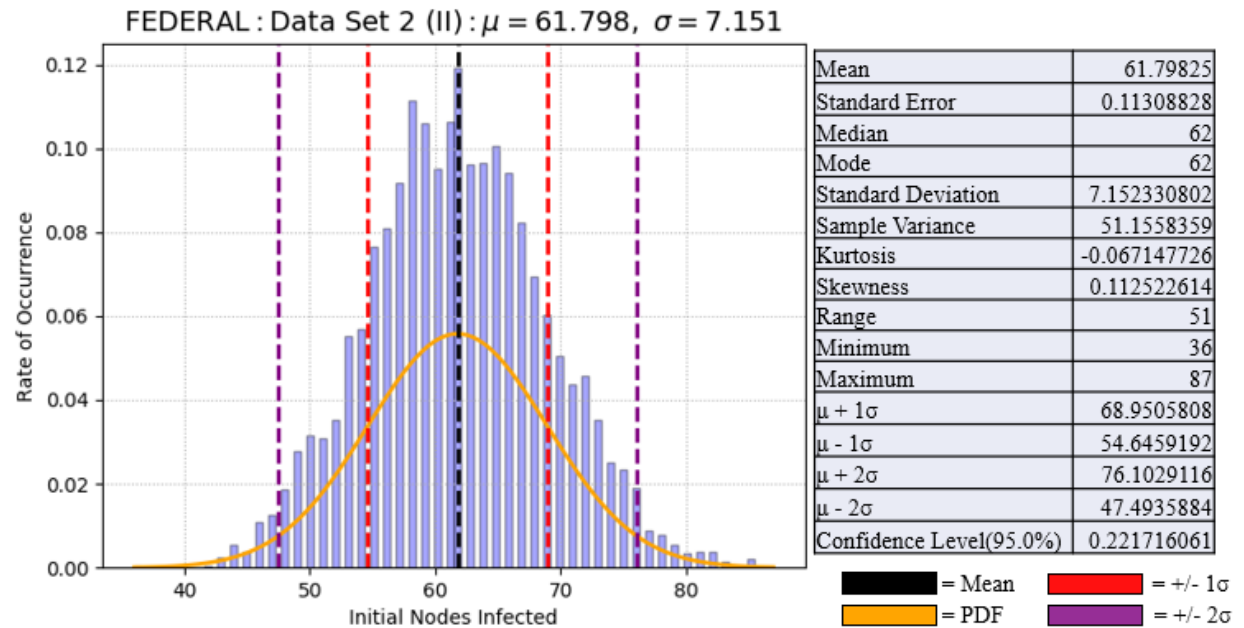
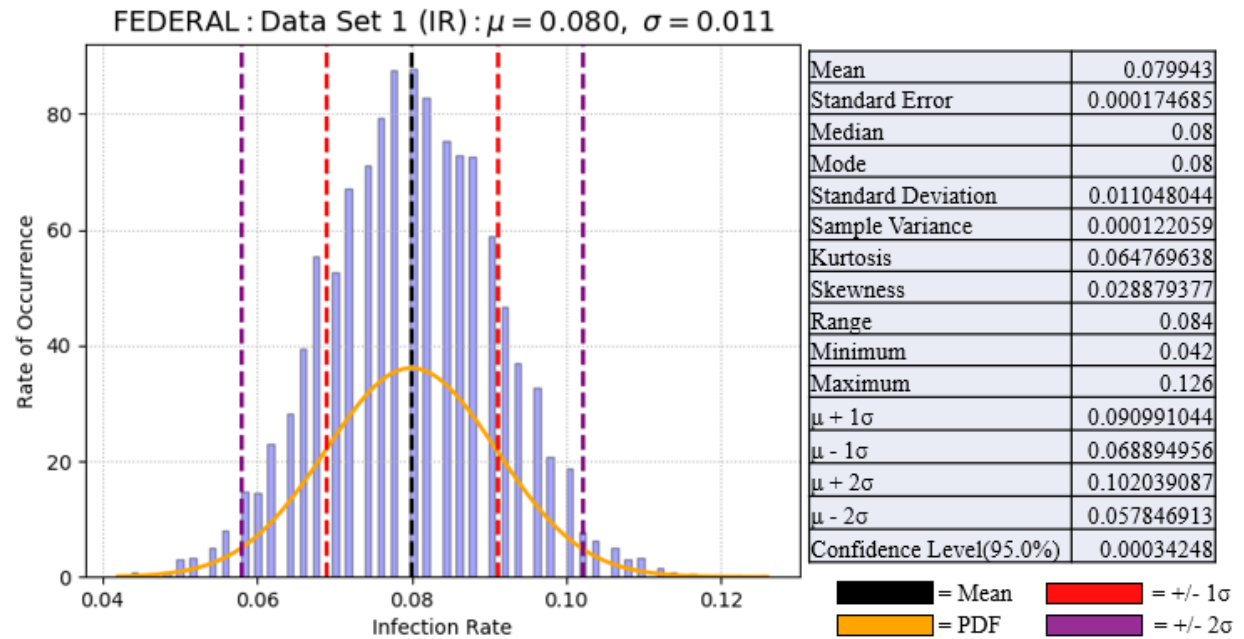




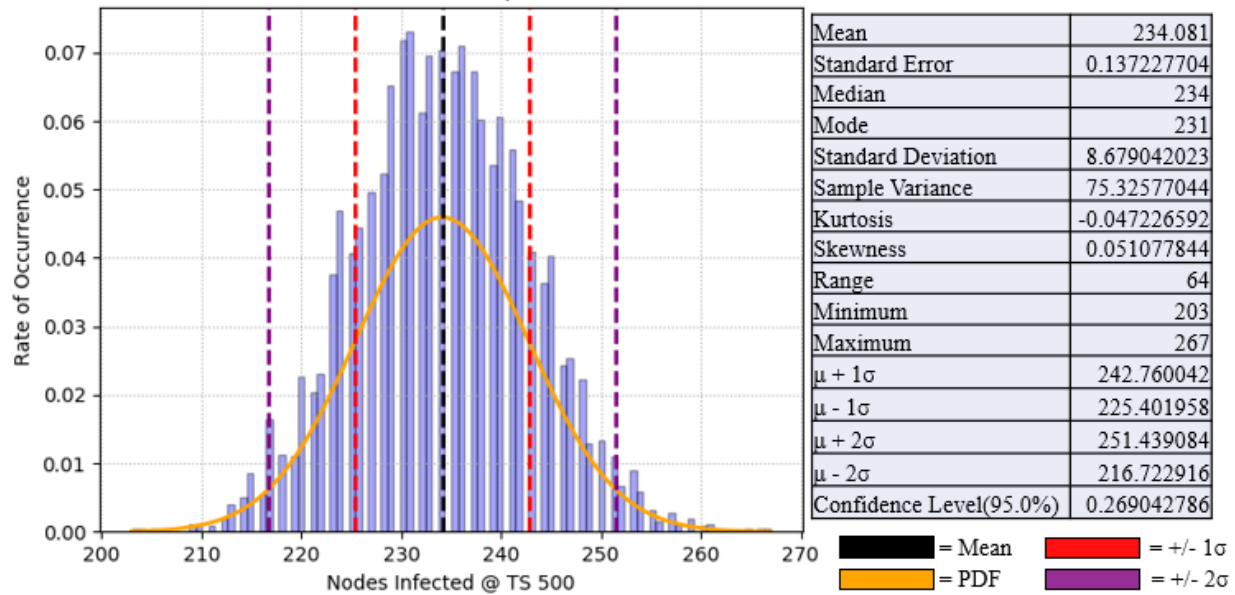


APPENDIX F. FEDERAL LEVEL NETWORK DATA GRAPHS AND STATISTICS

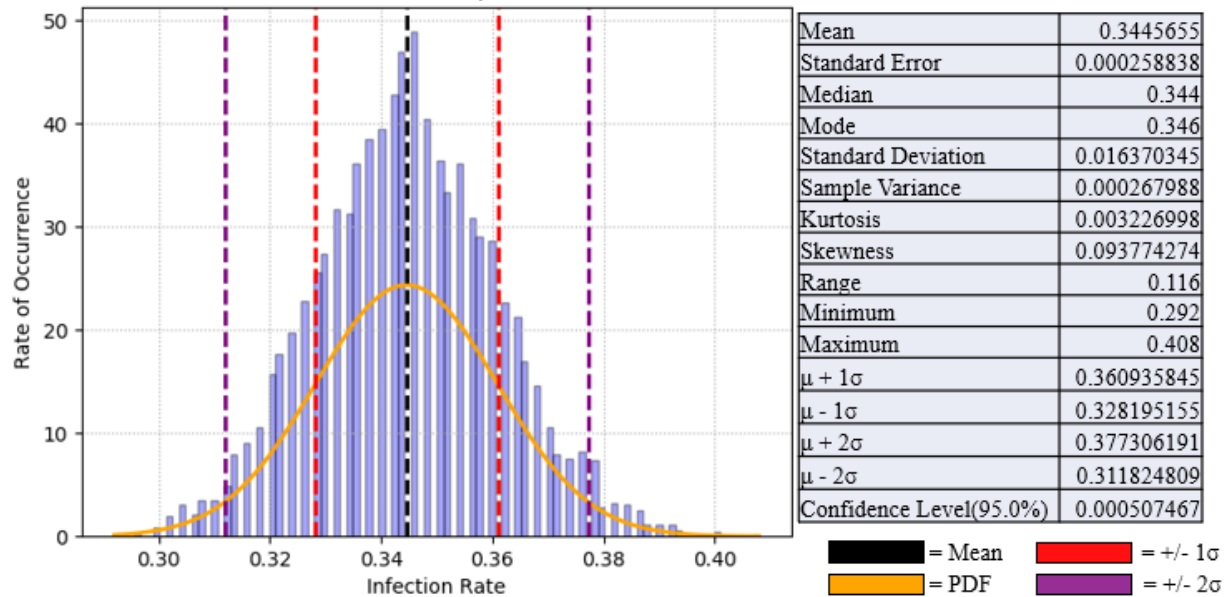


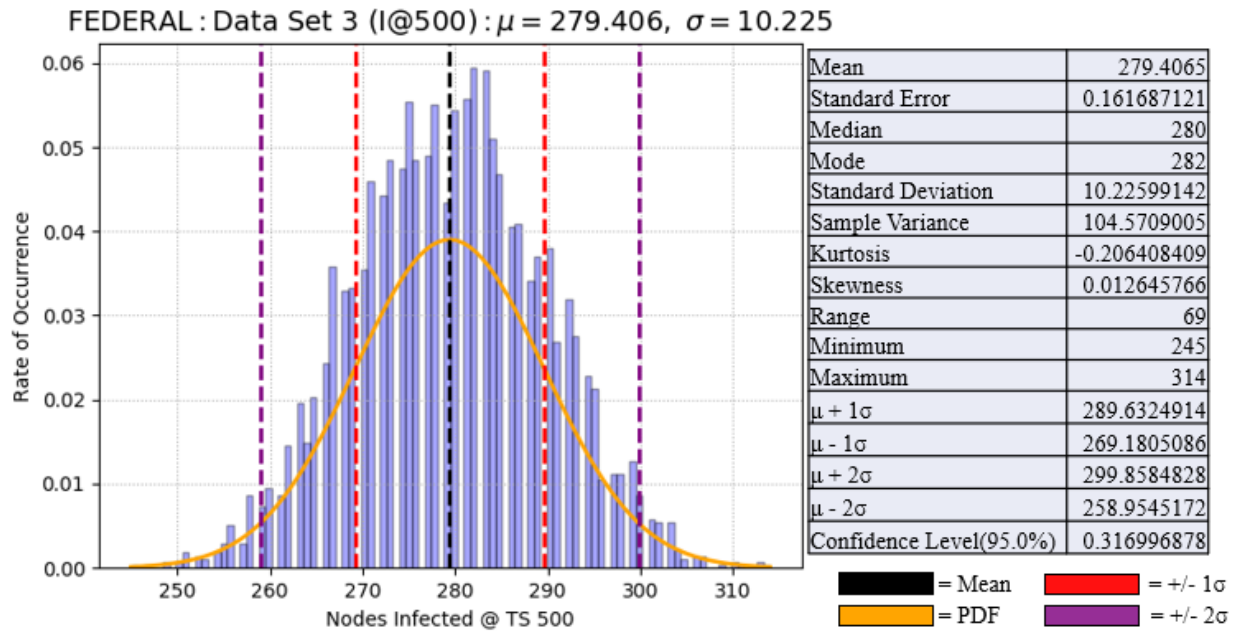
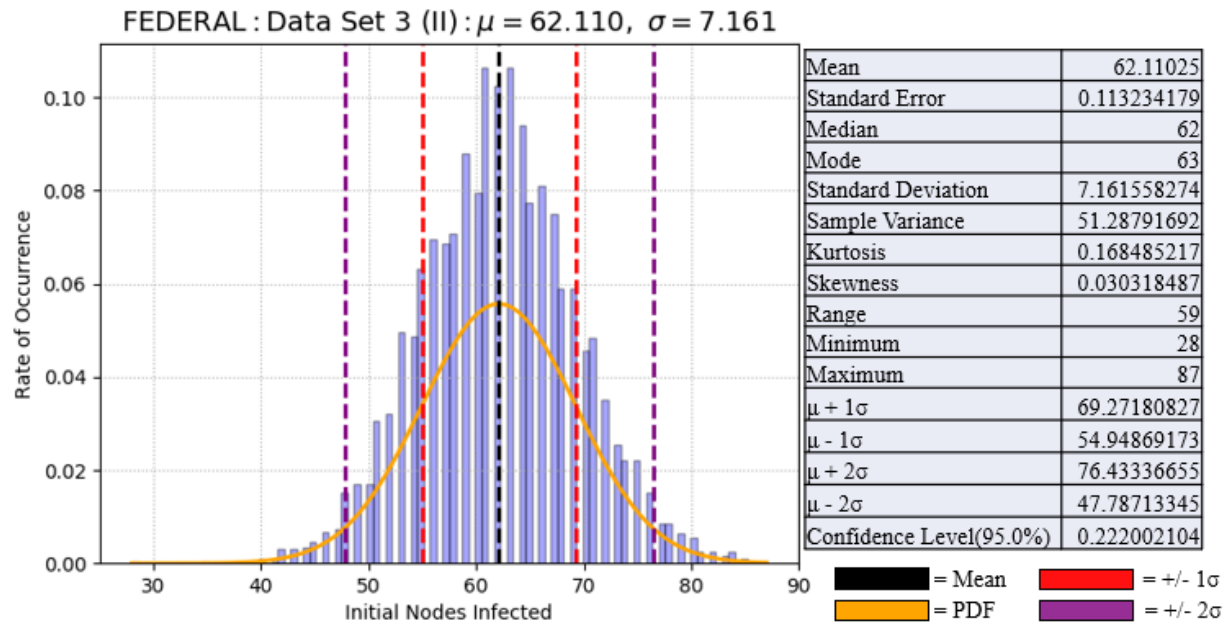


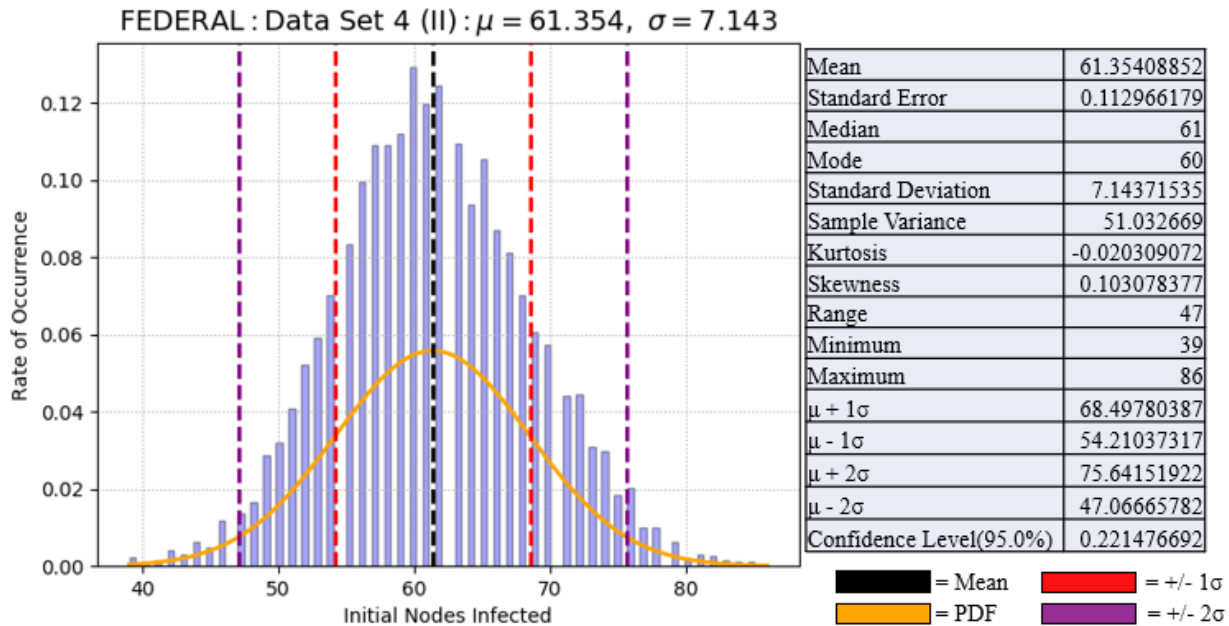
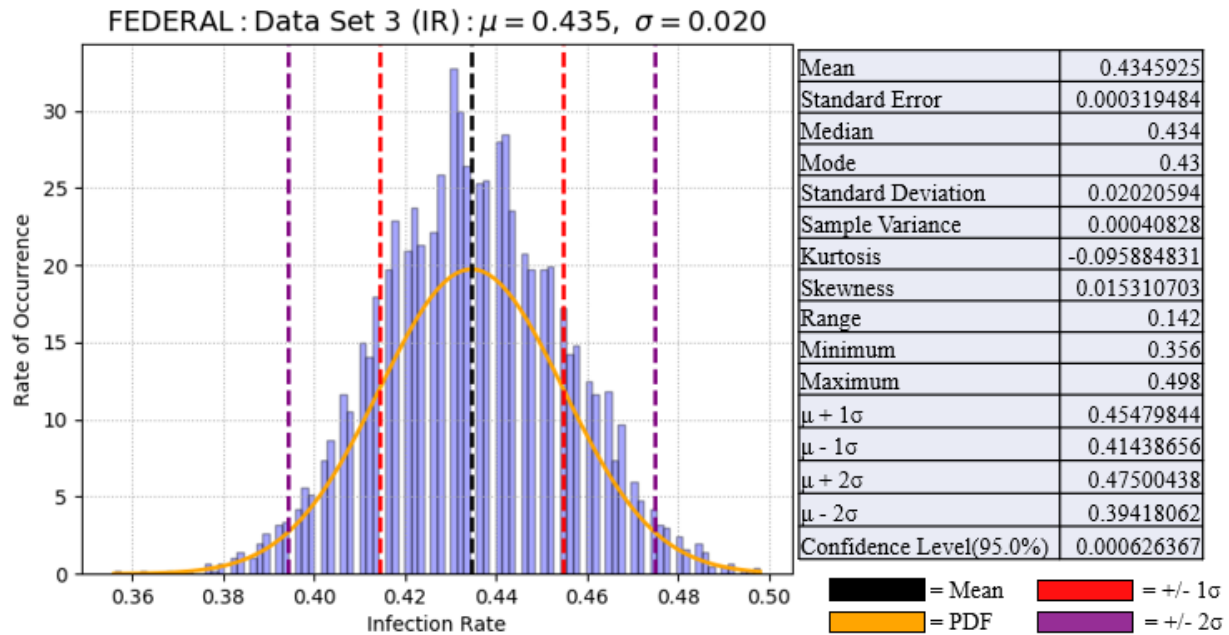
FEDERAL : Data Set 2 (I@500) : $\mu = 234.081$, $\sigma = 8.678$



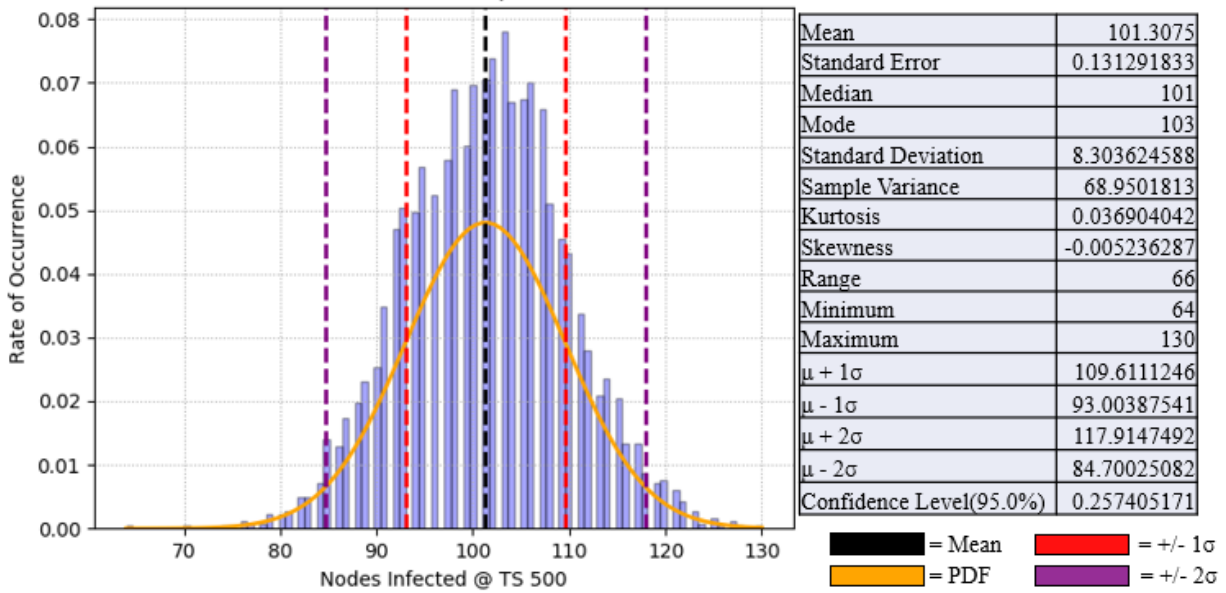
FEDERAL : Data Set 2 (IR) : $\mu = 0.345$, $\sigma = 0.016$



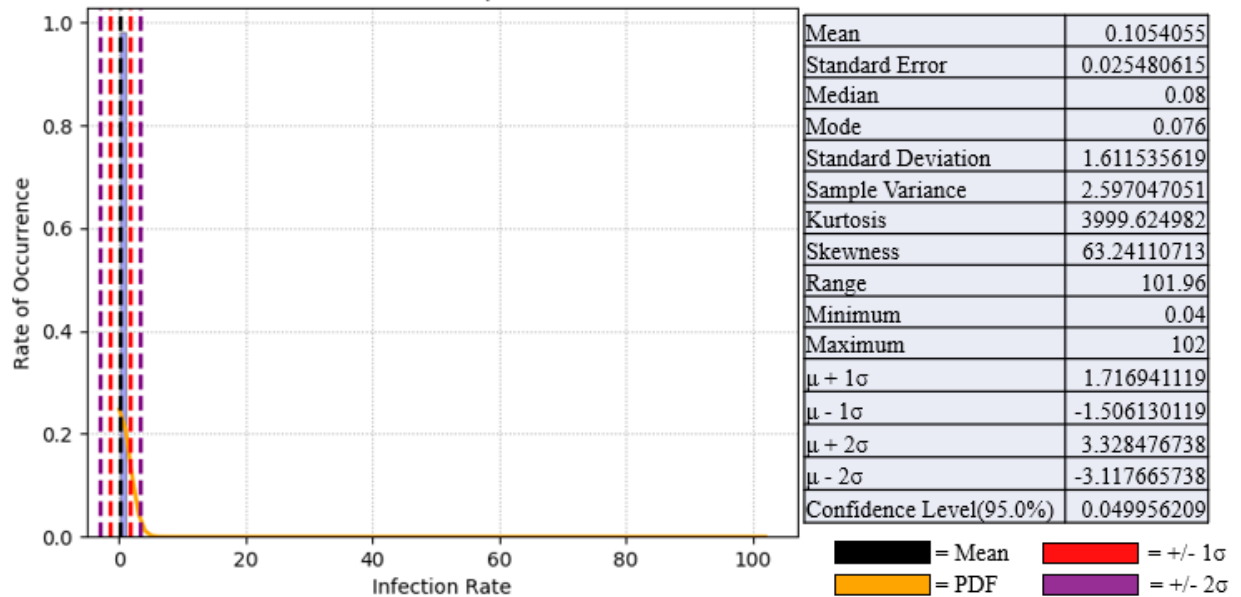


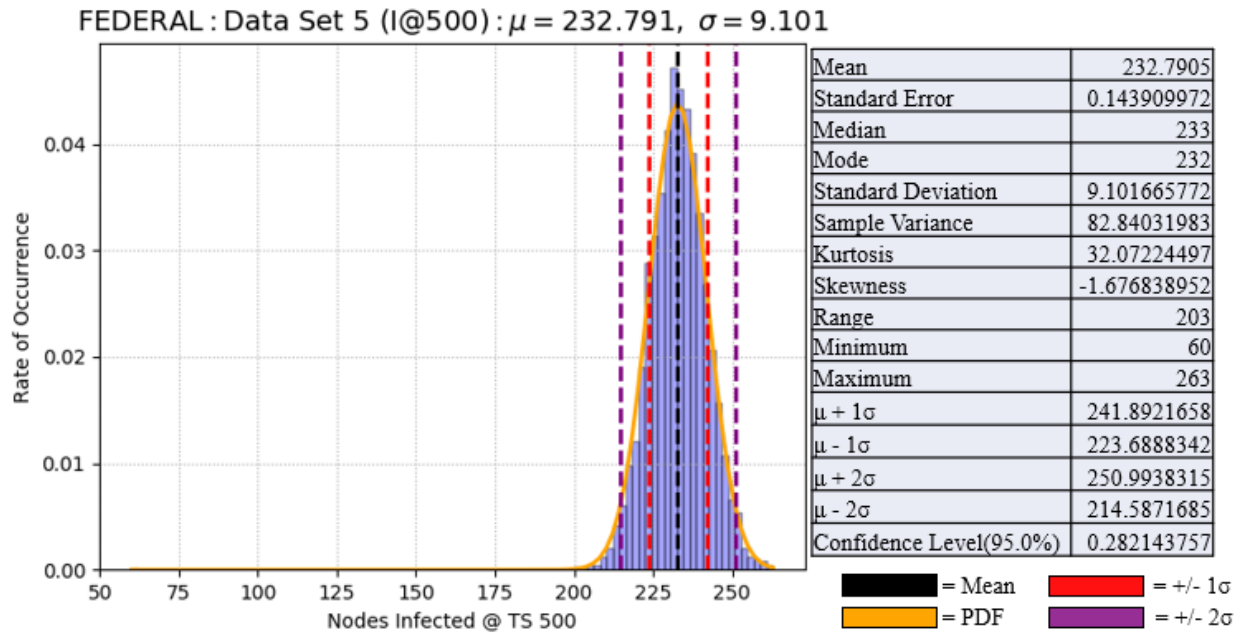
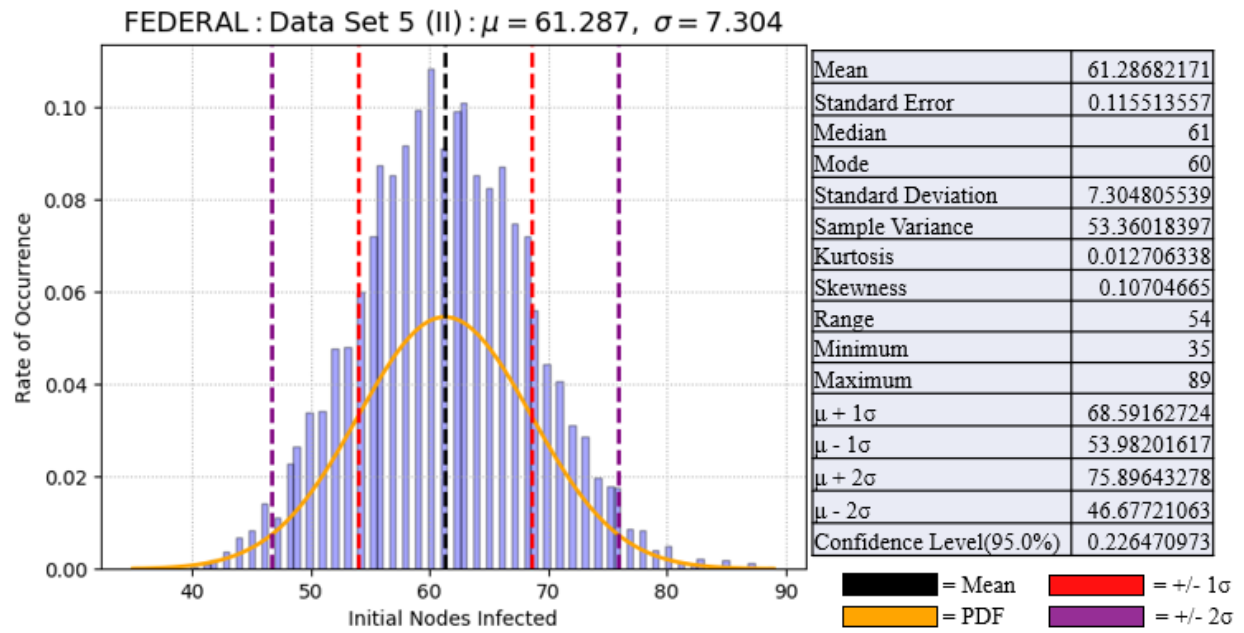


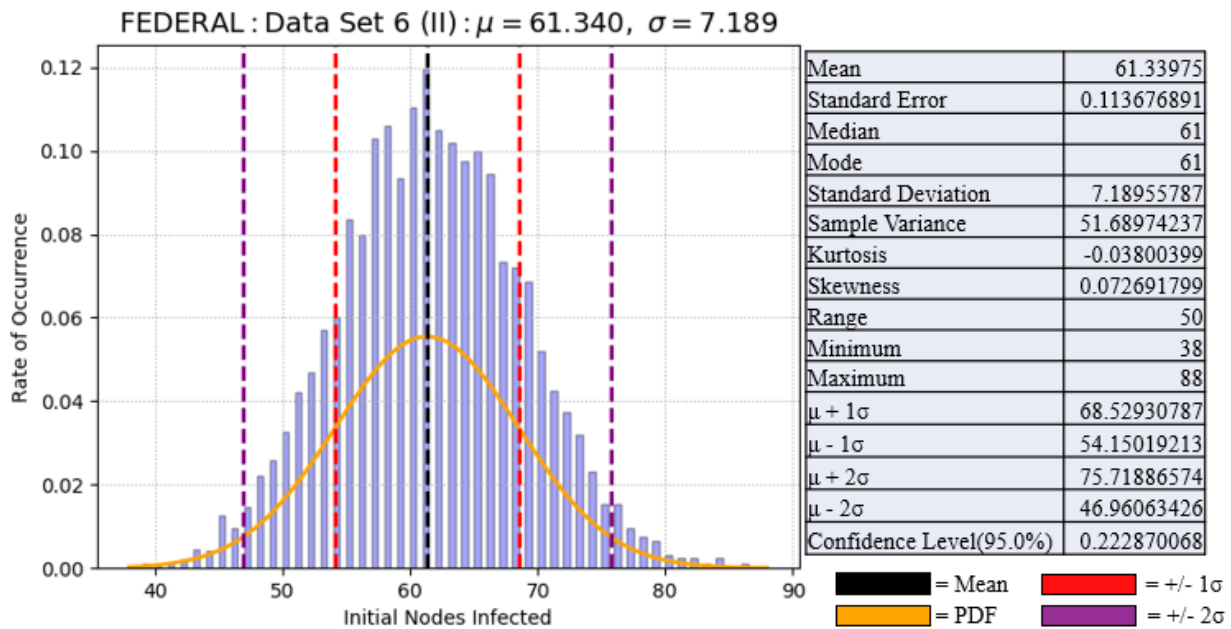
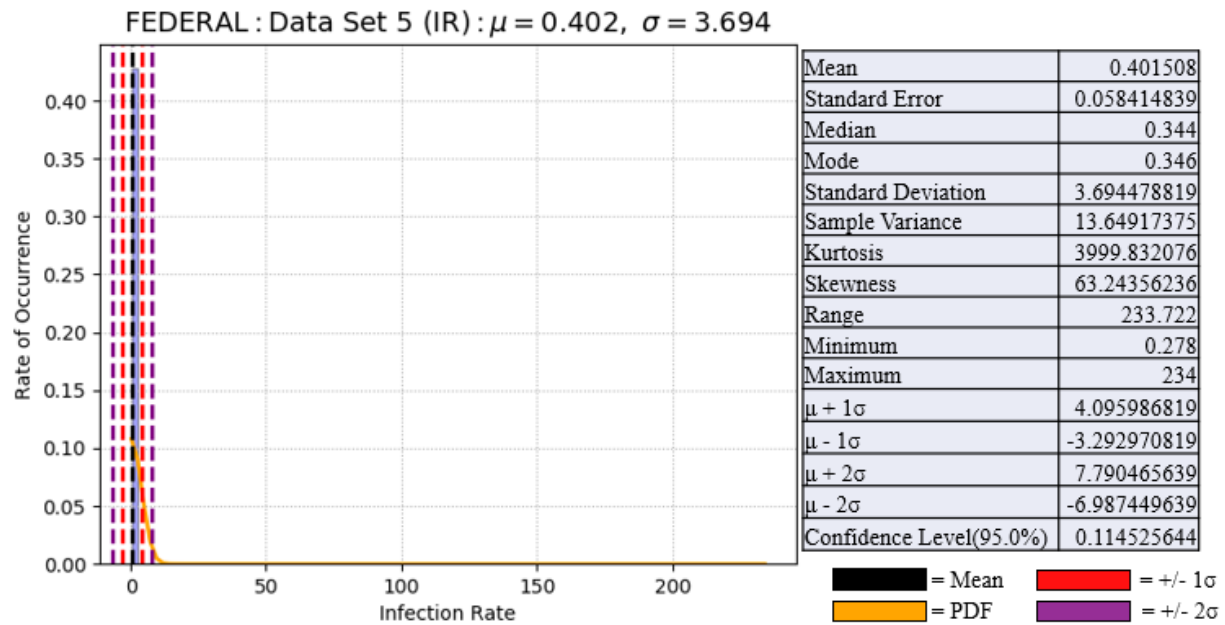
FEDERAL : Data Set 4 (I@500) : $\mu = 101.308$, $\sigma = 8.303$



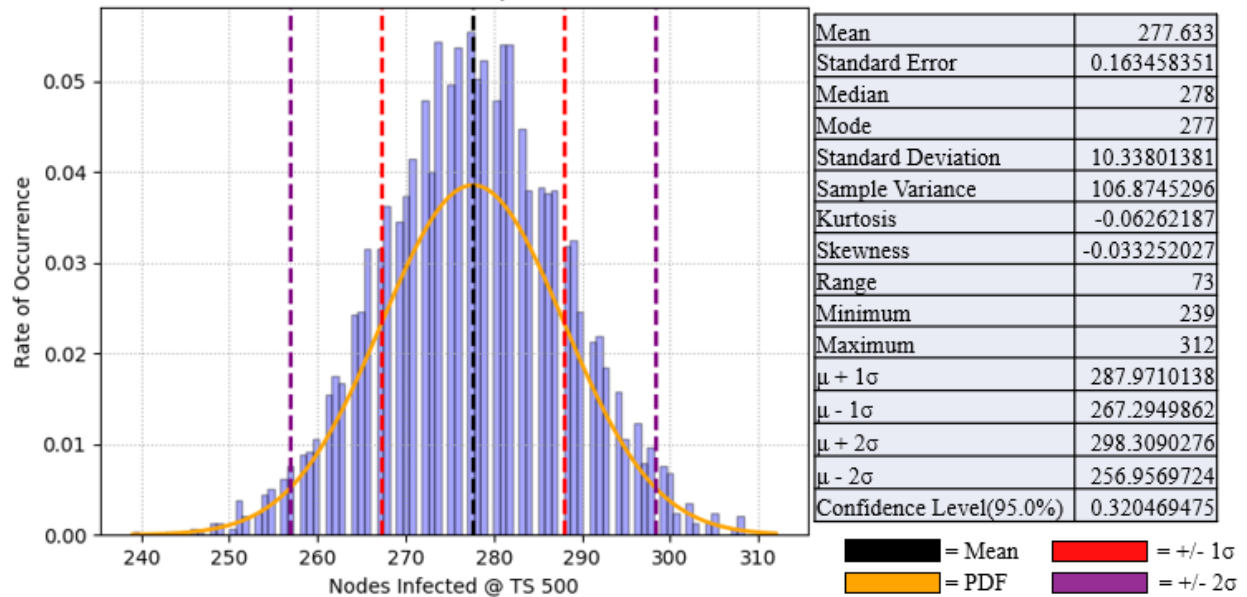
FEDERAL : Data Set 4 (IR) : $\mu = 0.105$, $\sigma = 1.611$



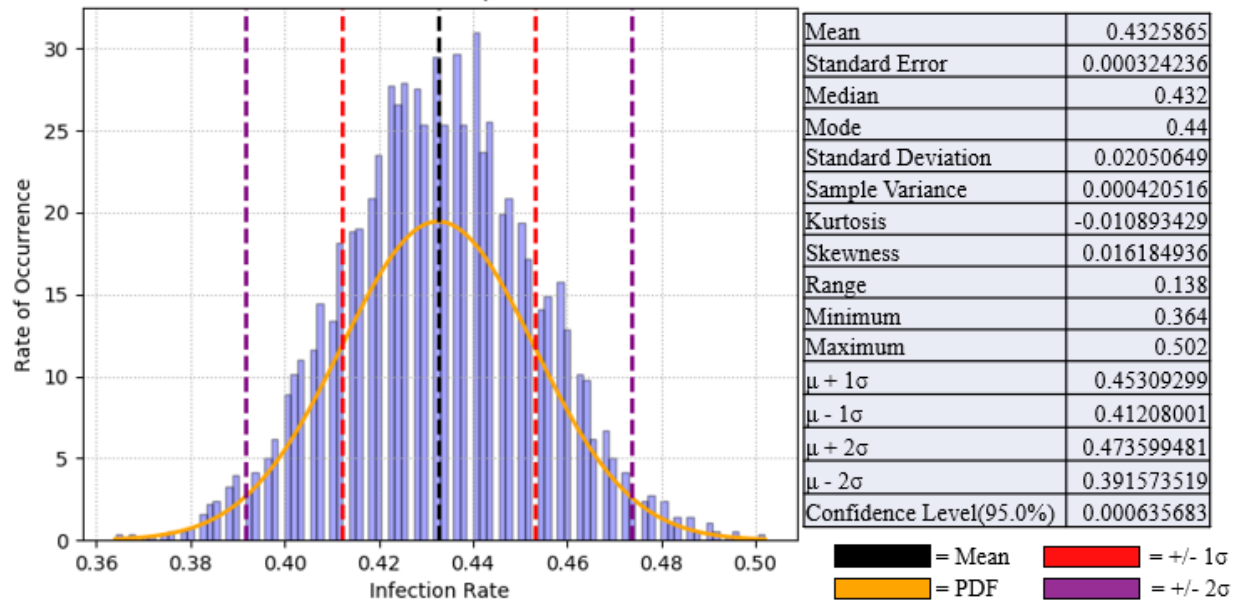


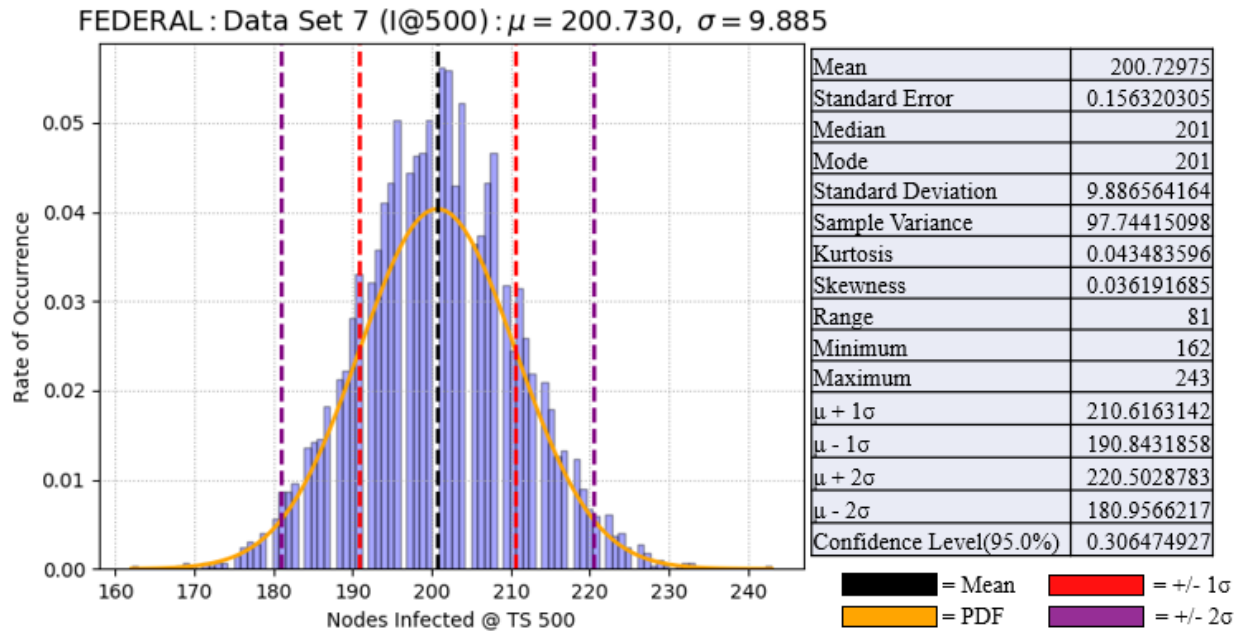
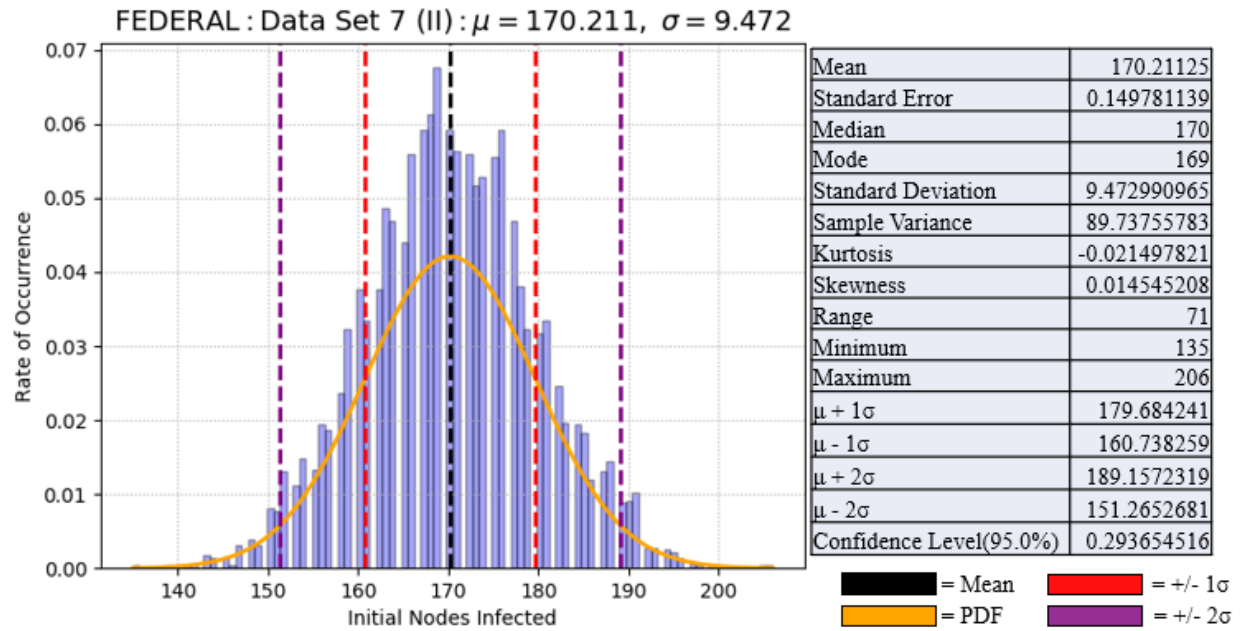


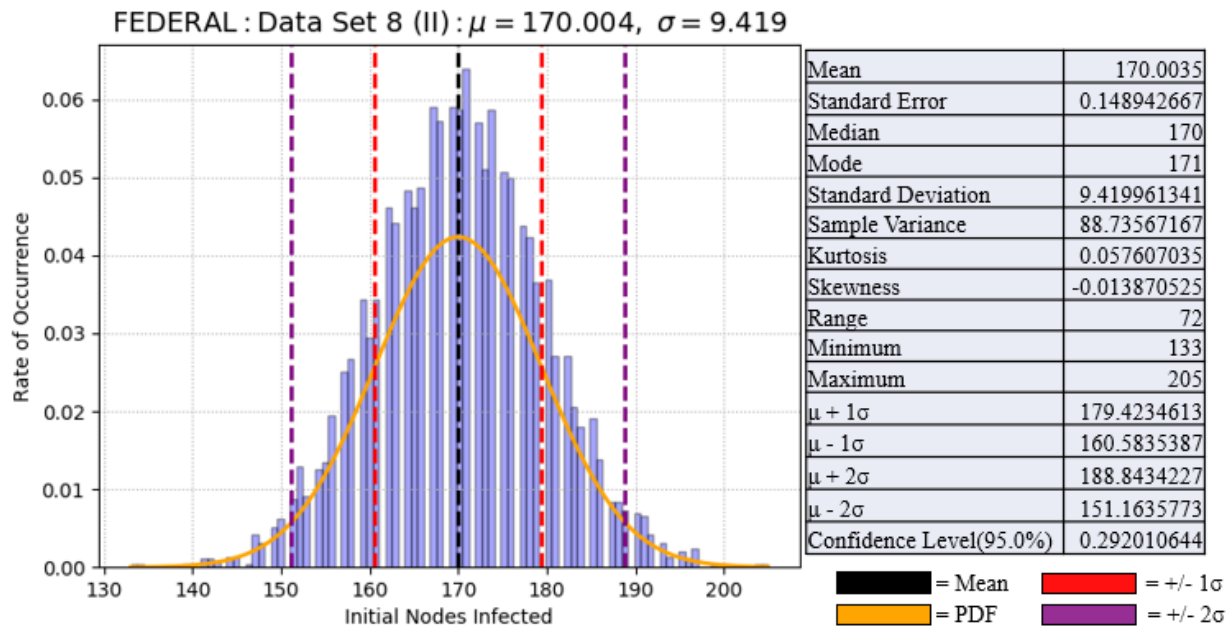
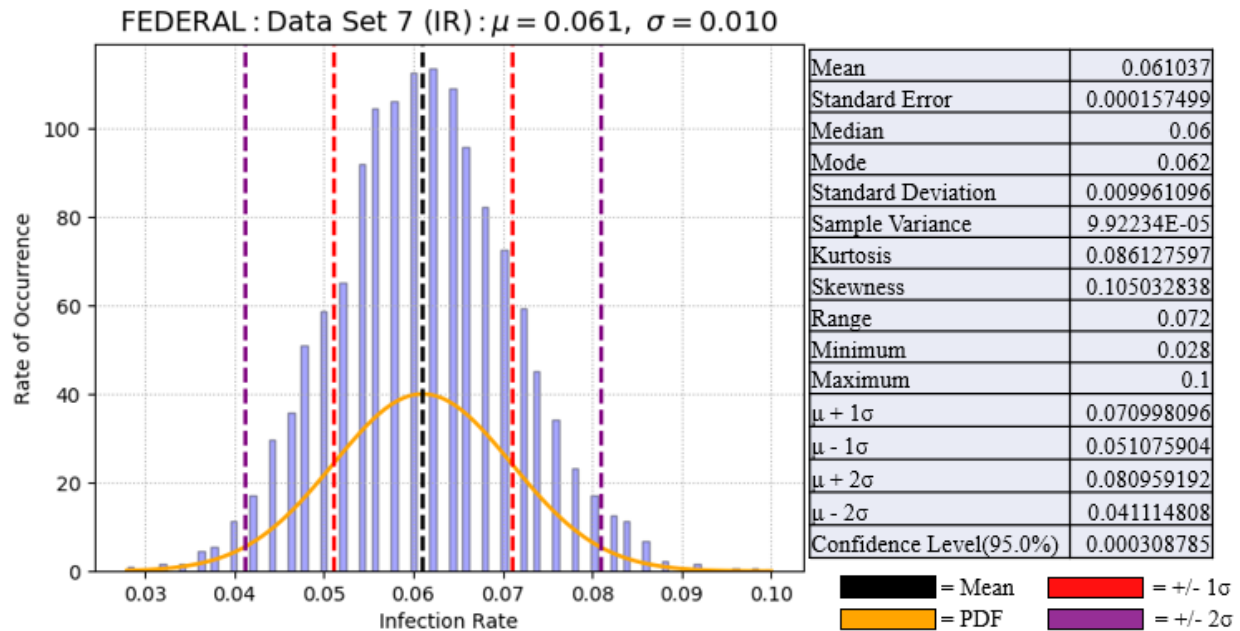
FEDERAL : Data Set 6 (I@500) : $\mu = 277.633$, $\sigma = 10.337$



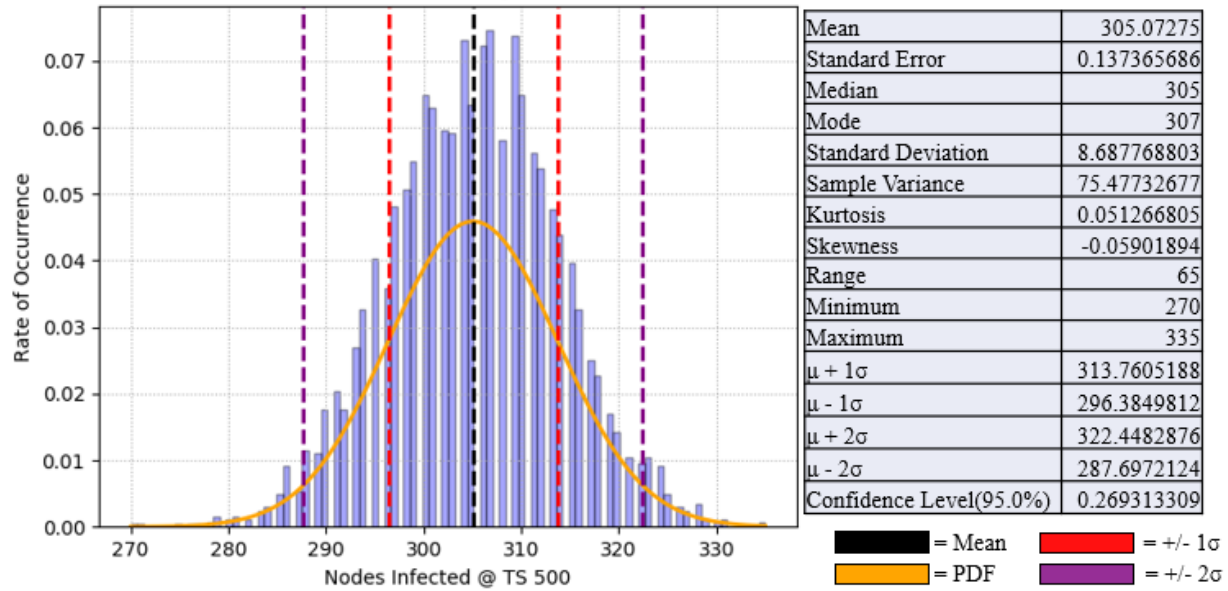
FEDERAL : Data Set 6 (IR) : $\mu = 0.433$, $\sigma = 0.021$



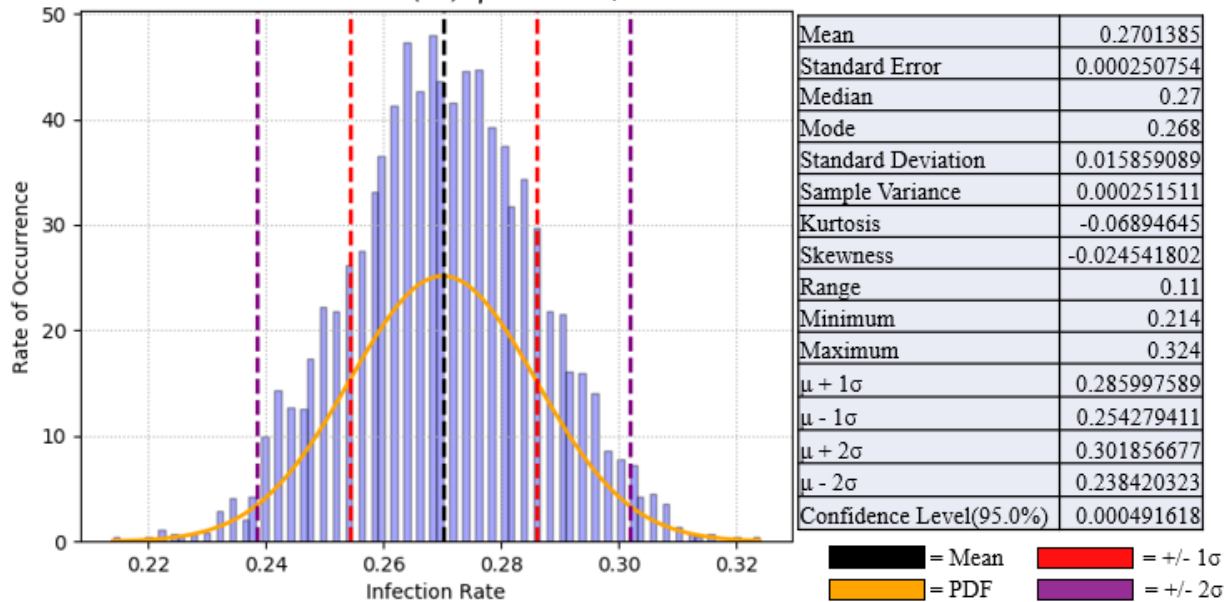


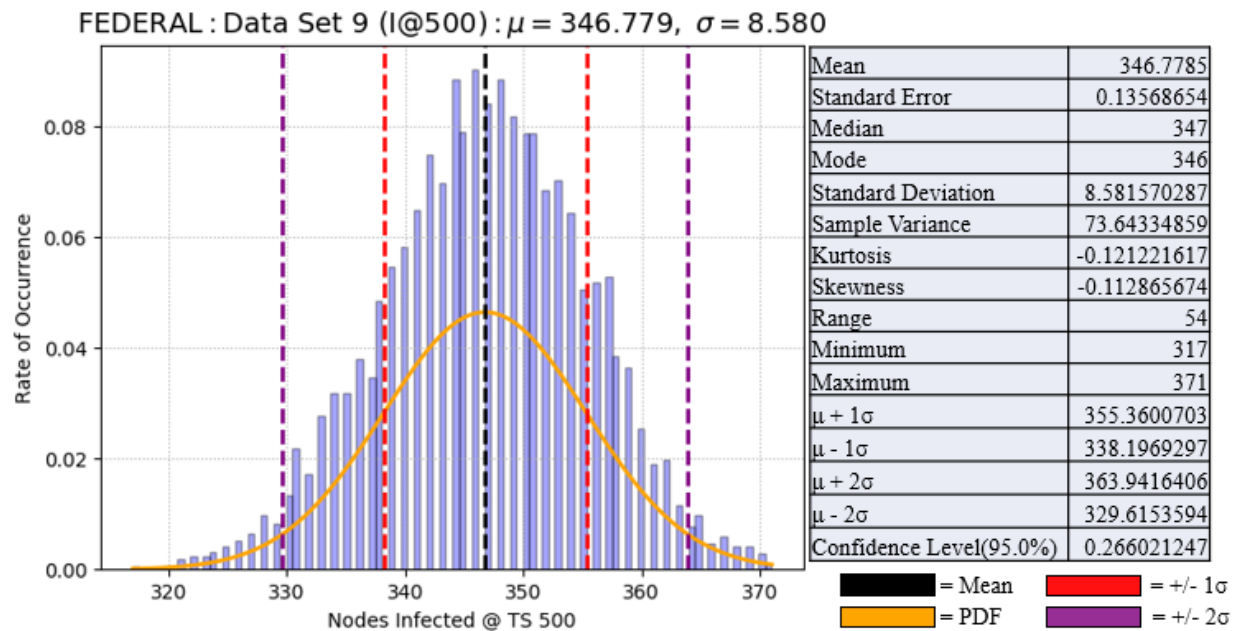
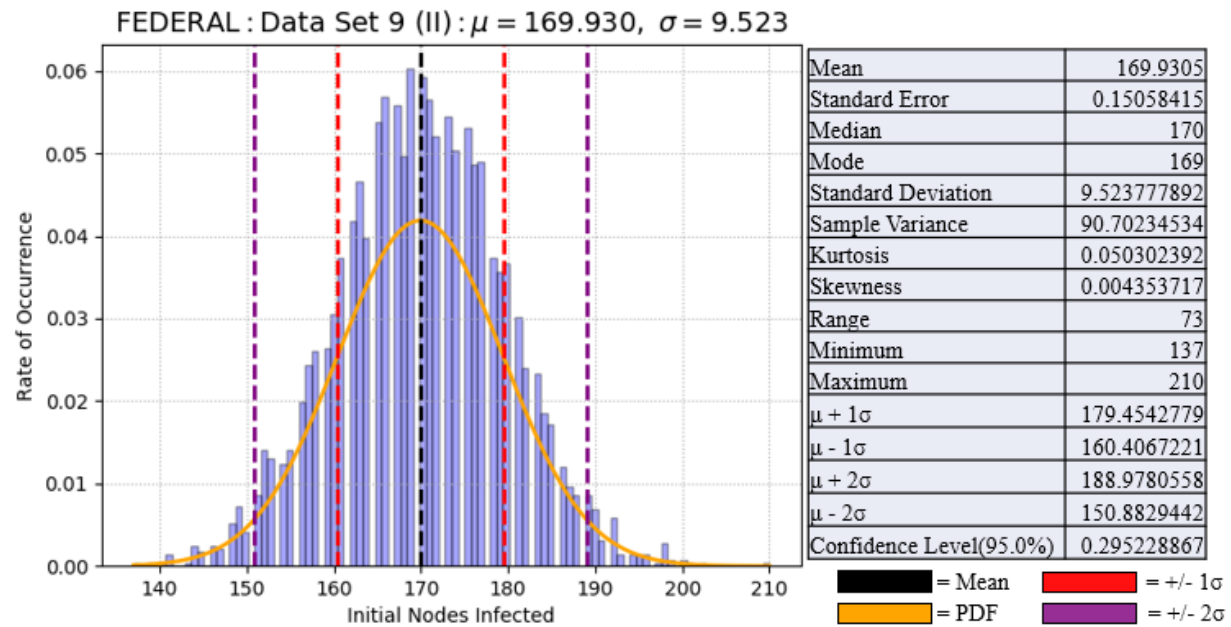


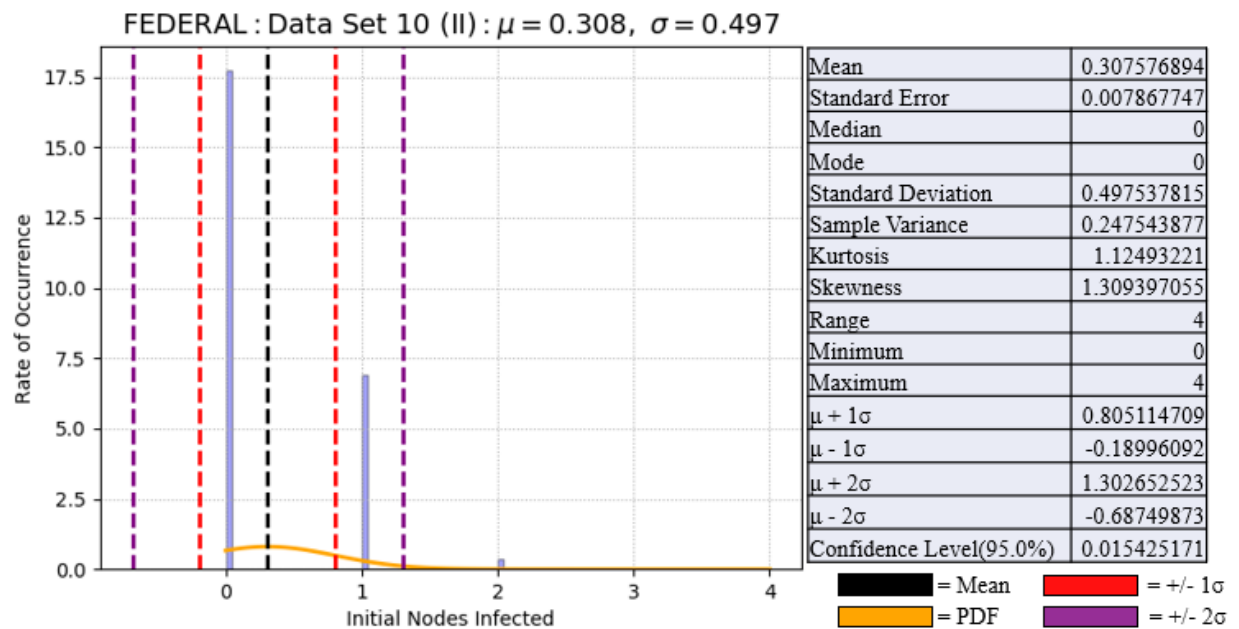
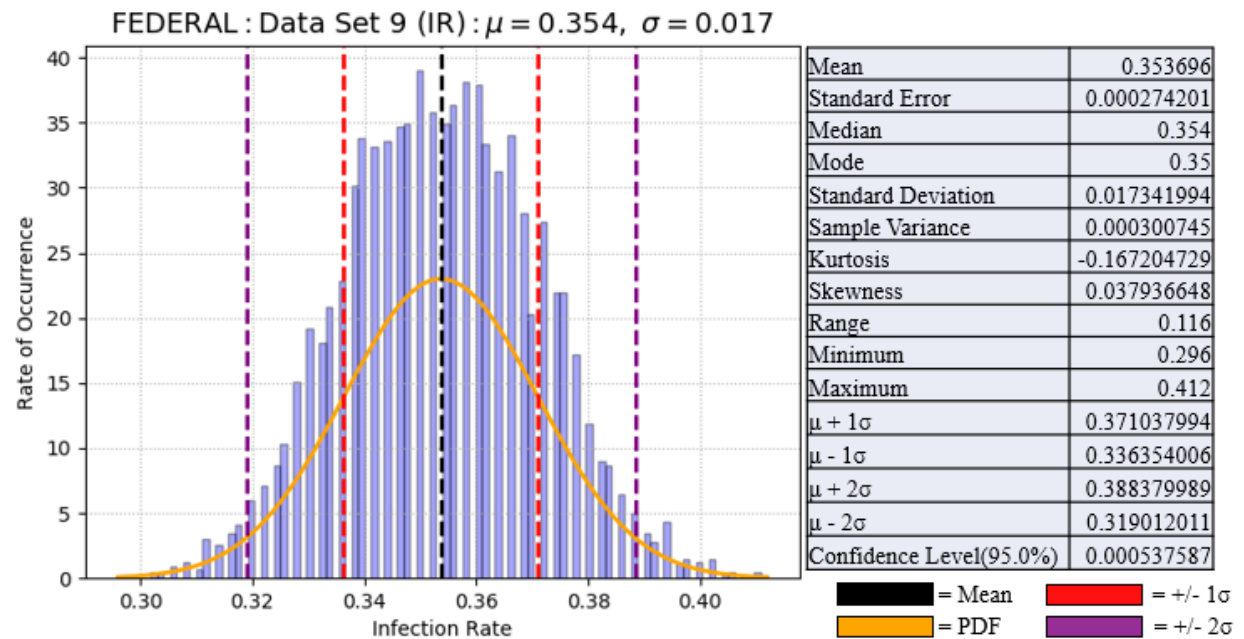
FEDERAL : Data Set 8 (I@500) : $\mu = 305.073$, $\sigma = 8.687$



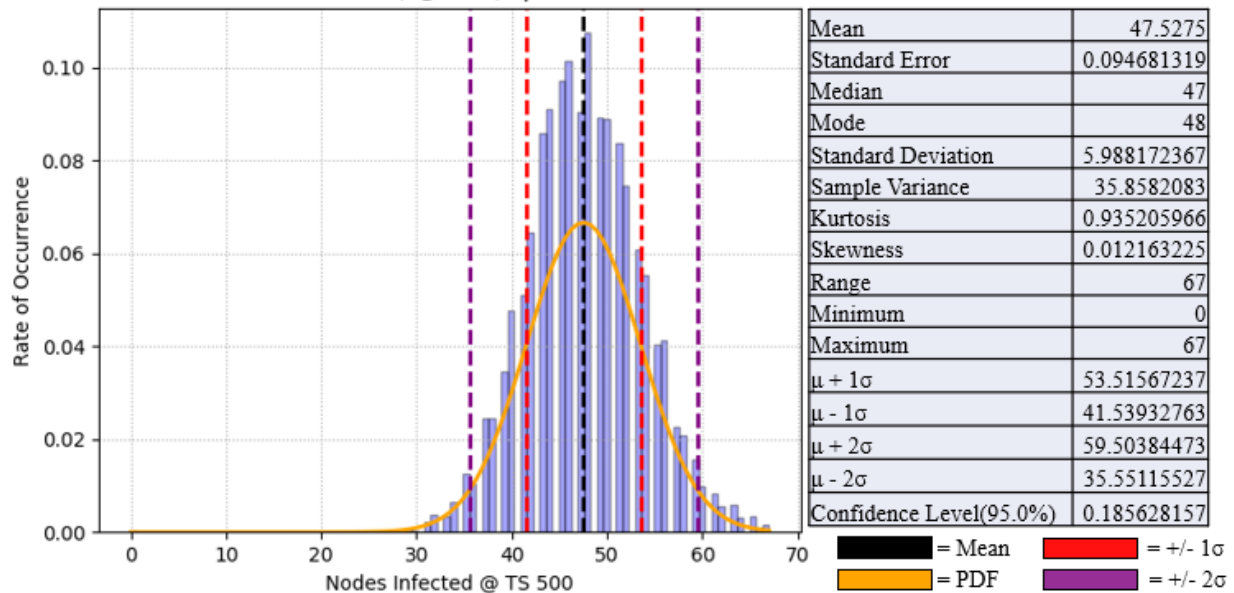
FEDERAL : Data Set 8 (IR) : $\mu = 0.270$, $\sigma = 0.016$



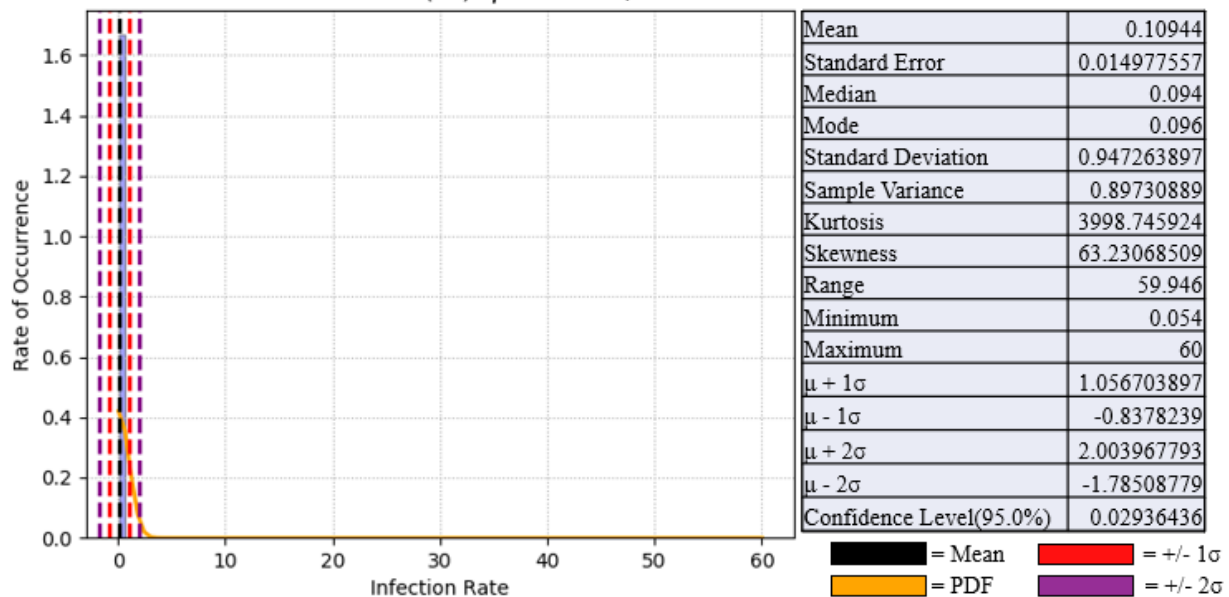




FEDERAL : Data Set 10 (I@500): $\mu = 47.528$, $\sigma = 5.987$

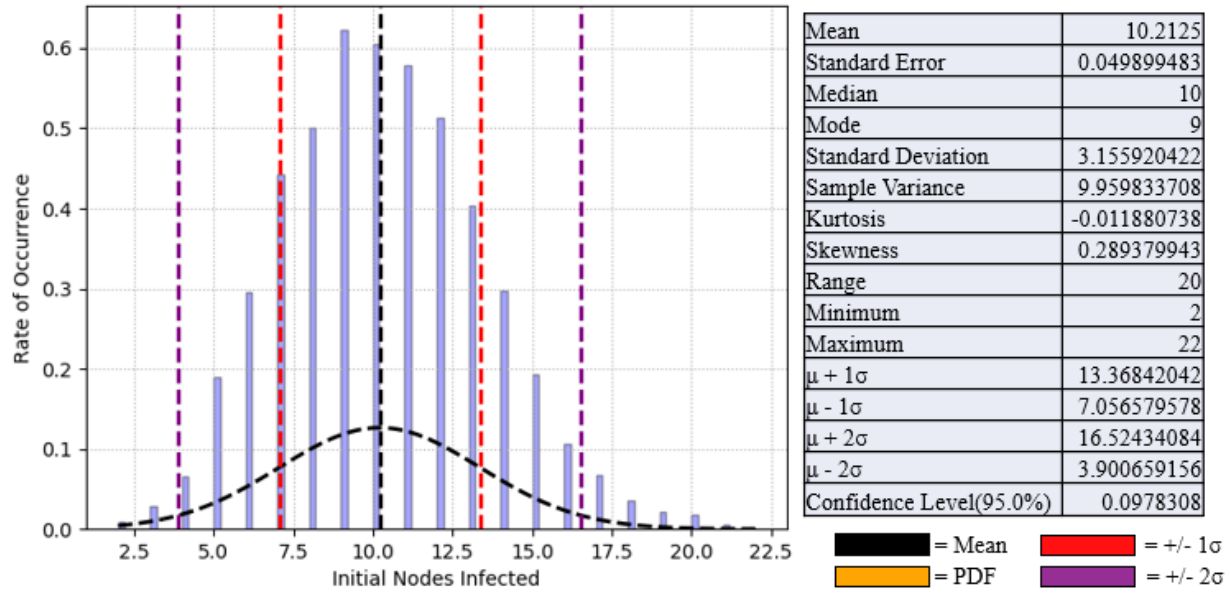


FEDERAL : Data Set 10 (IR): $\mu = 0.109$, $\sigma = 0.947$

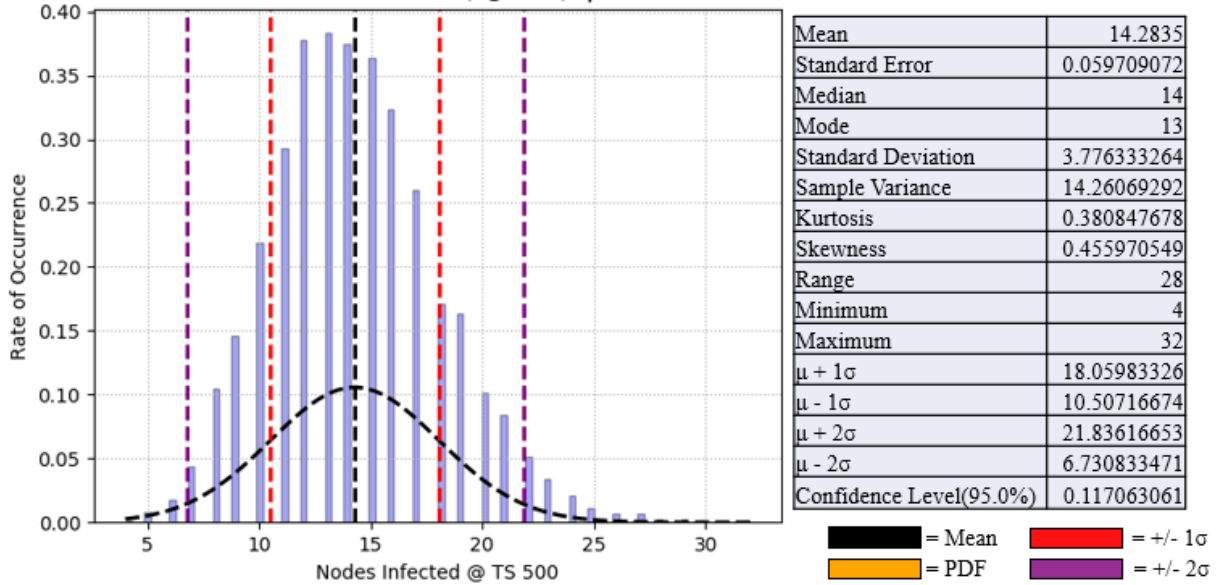


APPENDIX G. FEDERAL LEVEL WEB-BASED NETWORK DATA GRAPHS AND STATISTICS

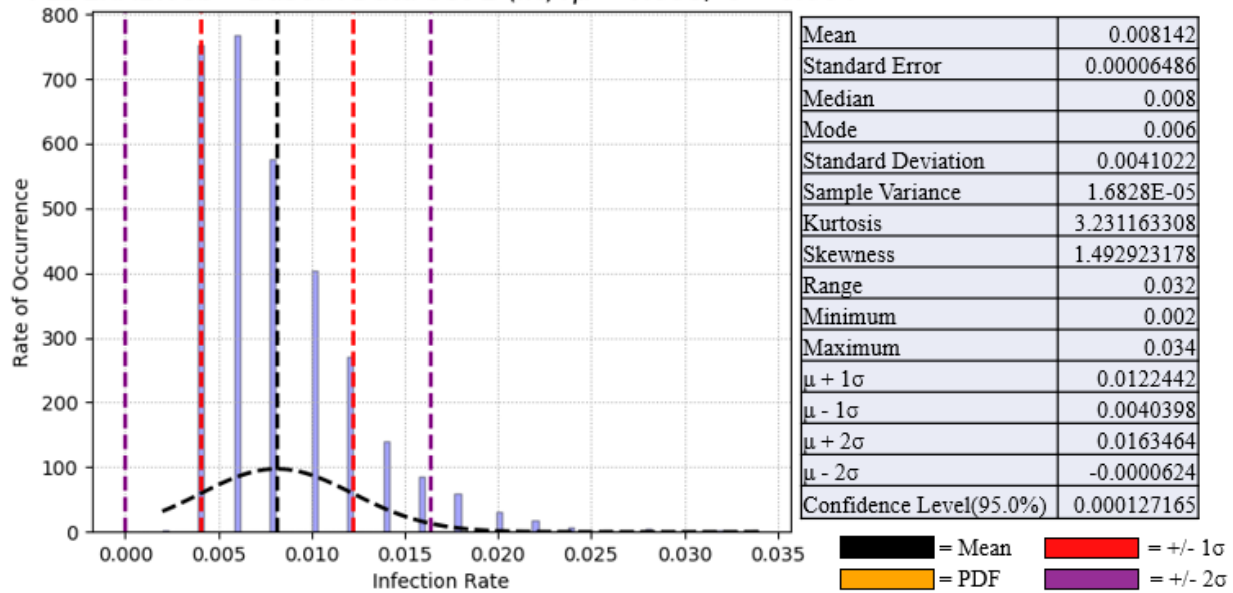
FEDERALWeb – Based : Data Set 1 (II) : $\mu = 10.213$, $\sigma = 3.156$



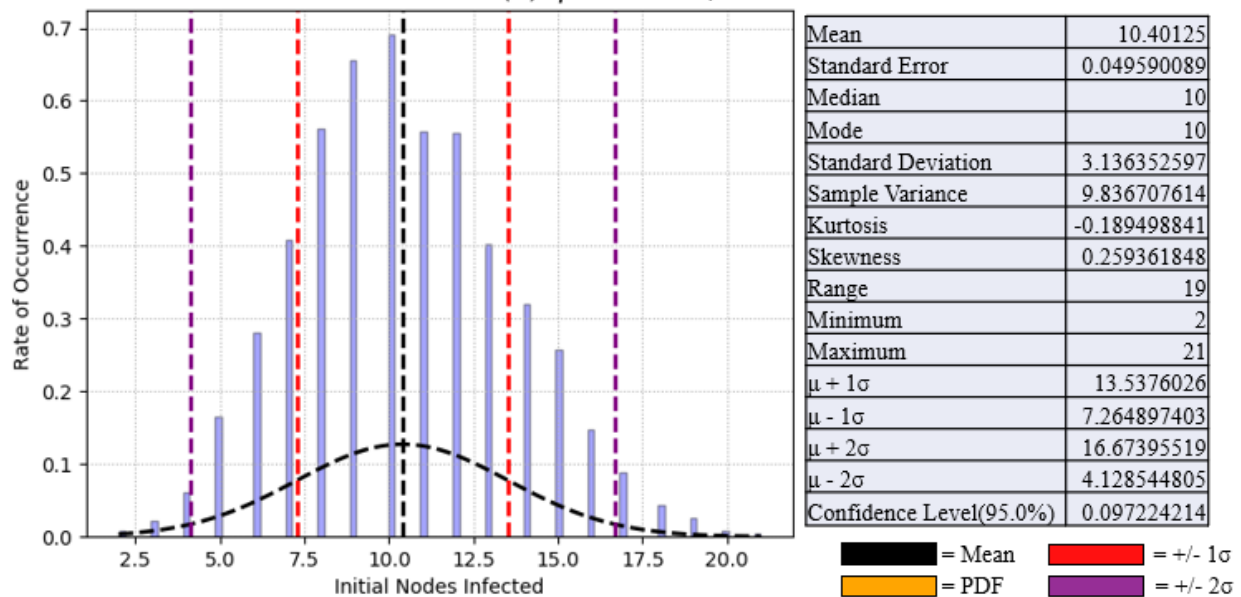
FEDERALWeb – Based : Data Set 1 (I@500) : $\mu = 14.284$, $\sigma = 3.771$



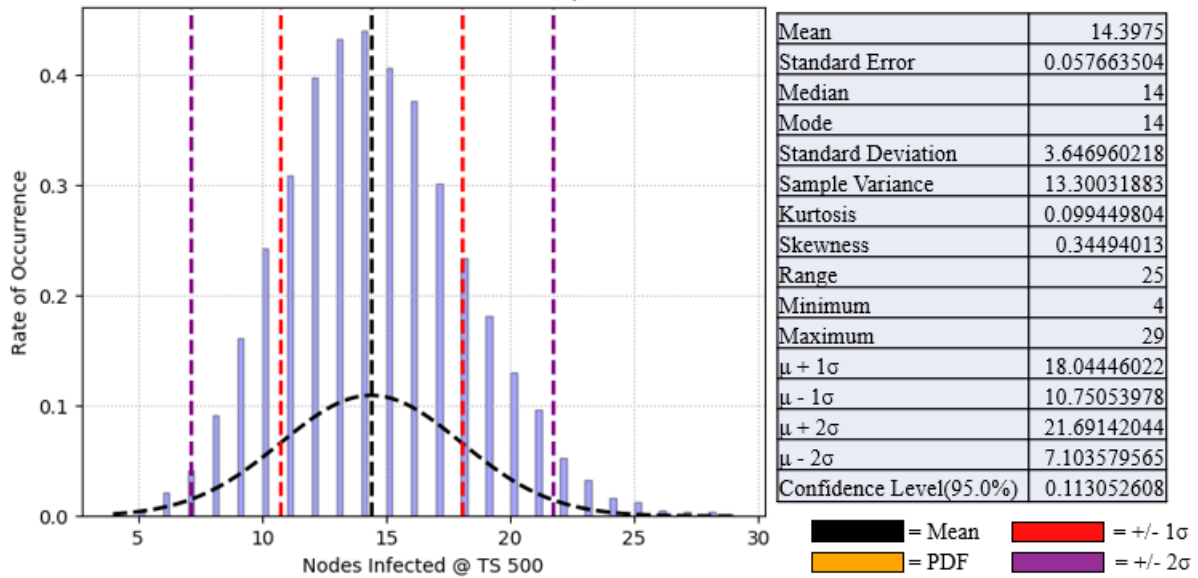
FEDERALWeb – Based : Data Set 1 (IR) : $\mu = 0.008$, $\sigma = 0.004$



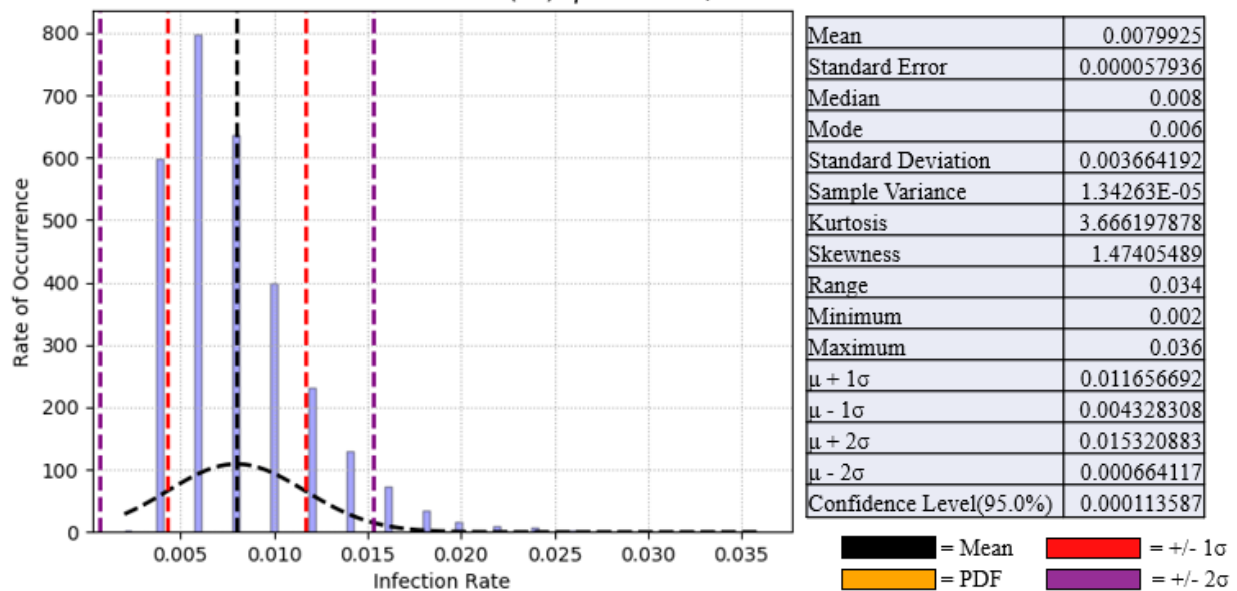
FEDERALWeb – Based : Data Set 2 (II) : $\mu = 10.401$, $\sigma = 3.136$



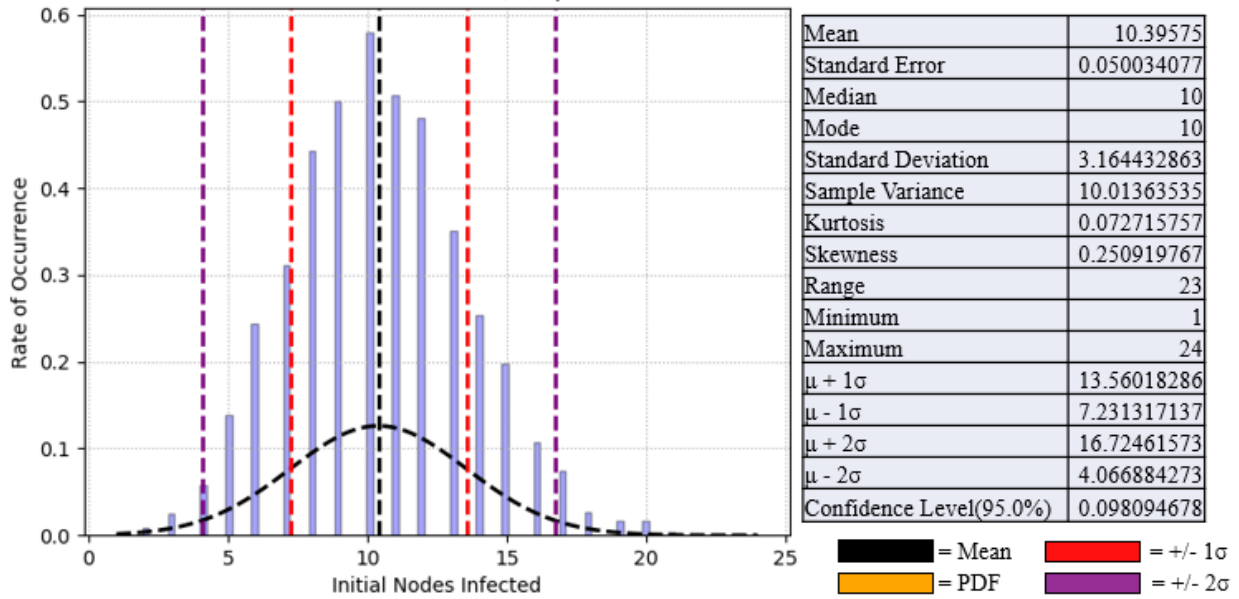
FEDERALWeb – Based : Data Set 2 (I@500) : $\mu = 14.398$, $\sigma = 3.64$



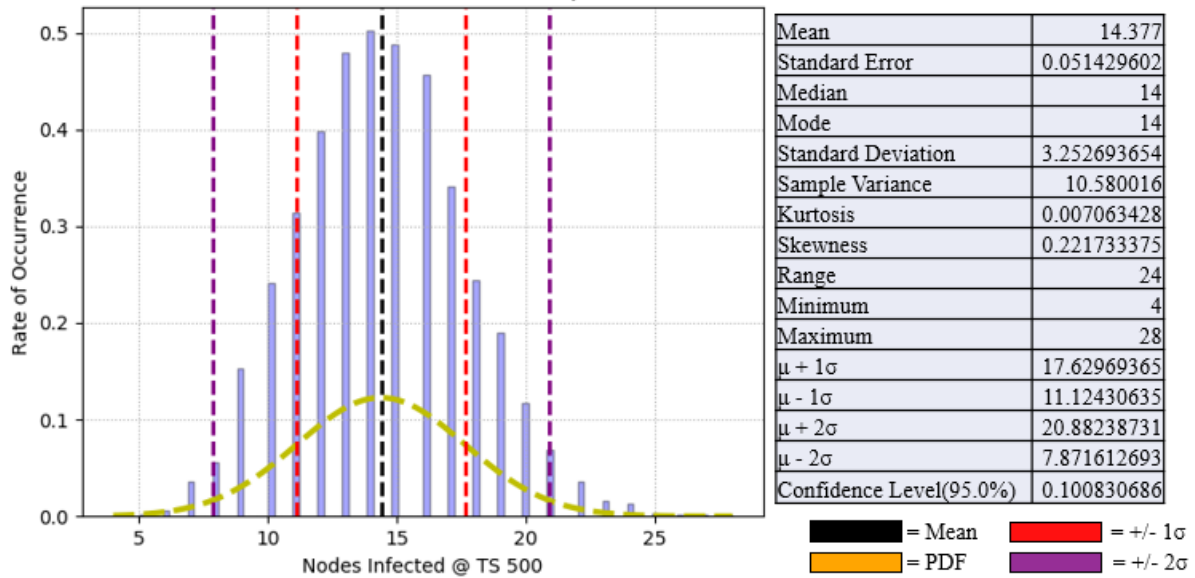
FEDERALWeb – Based : Data Set 2 (IR) : $\mu = 0.008$, $\sigma = 0.004$



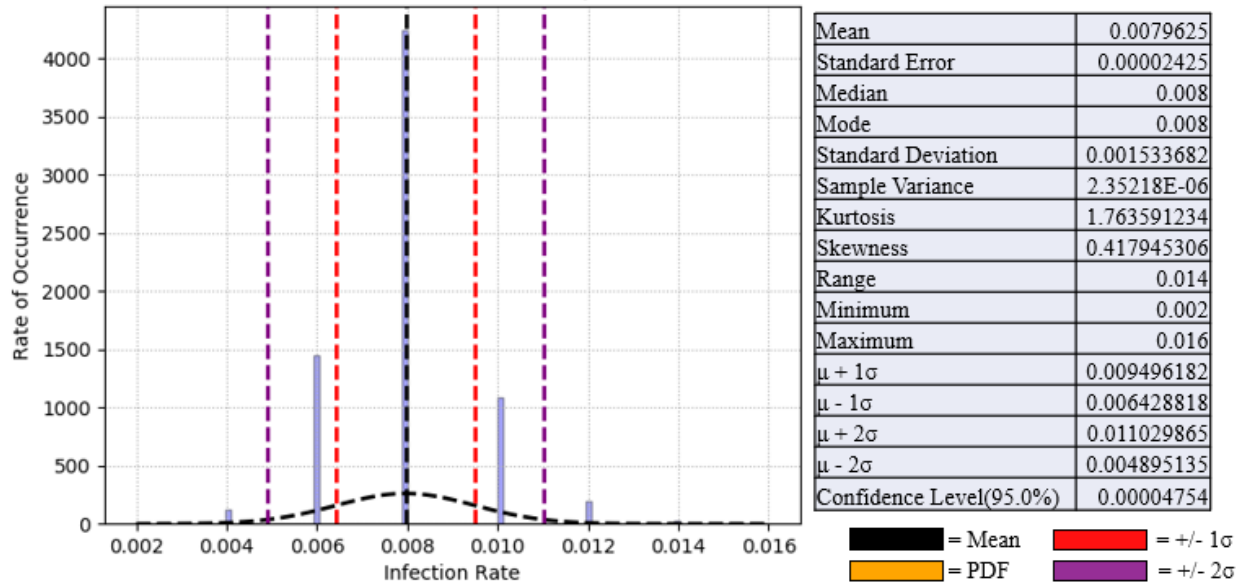
FEDERALWeb – Based : Data Set 3 (II) : $\mu = 10.396$, $\sigma = 3.164$



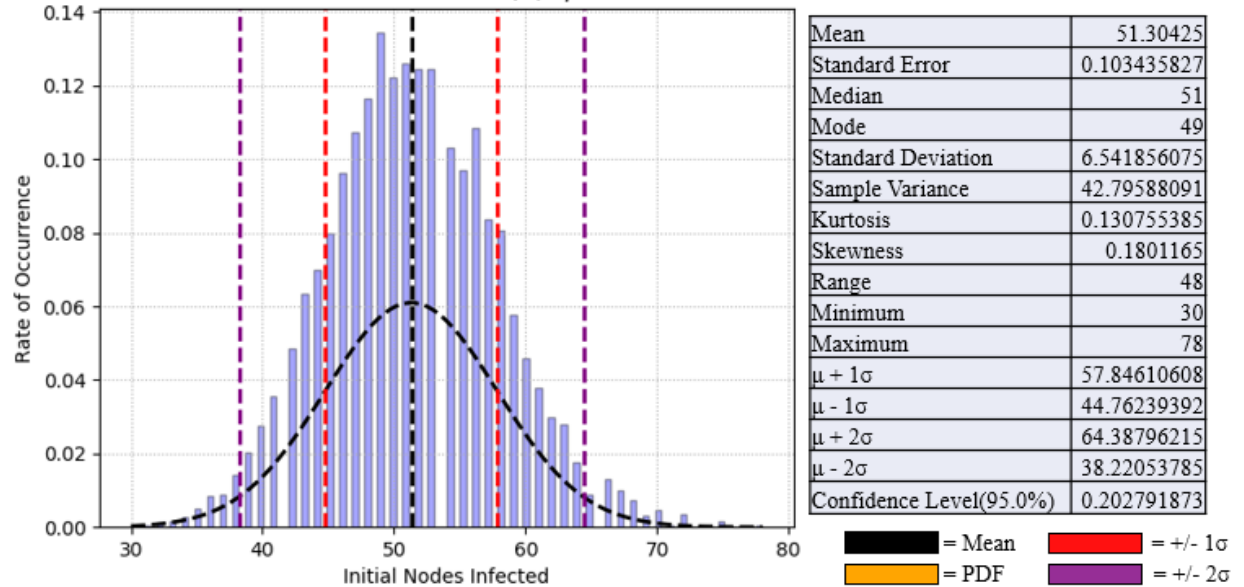
FEDERALWeb – Based : Data Set 3 (I@500) : $\mu = 14.377$, $\sigma = 3.25$



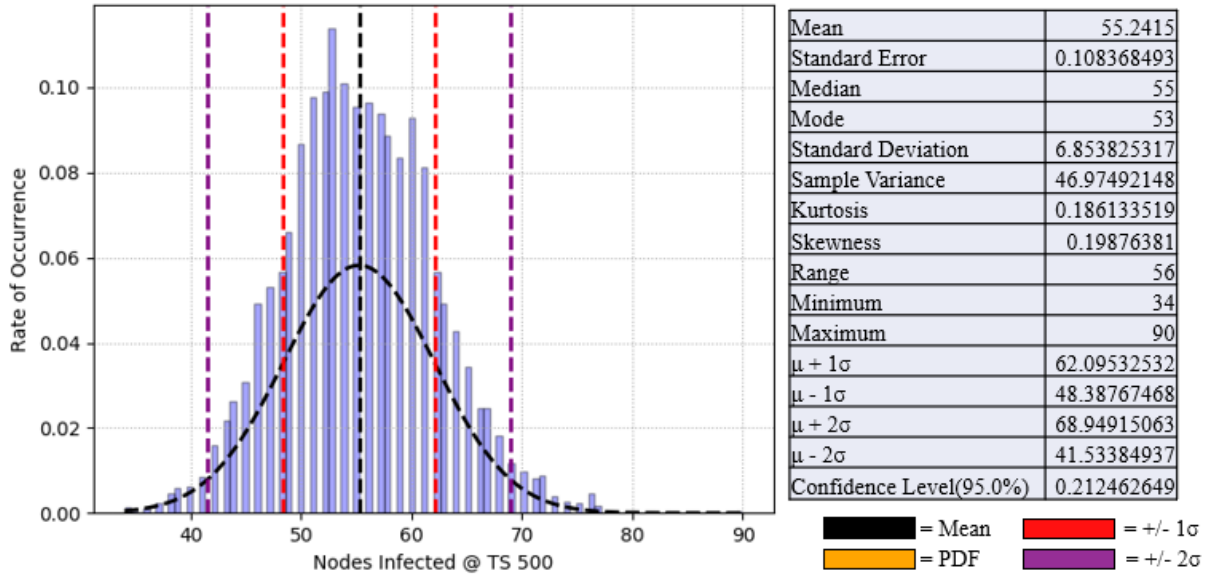
FEDERALWeb – Based : Data Set 3 (IR) : $\mu = 0.008$, $\sigma = 0.002$



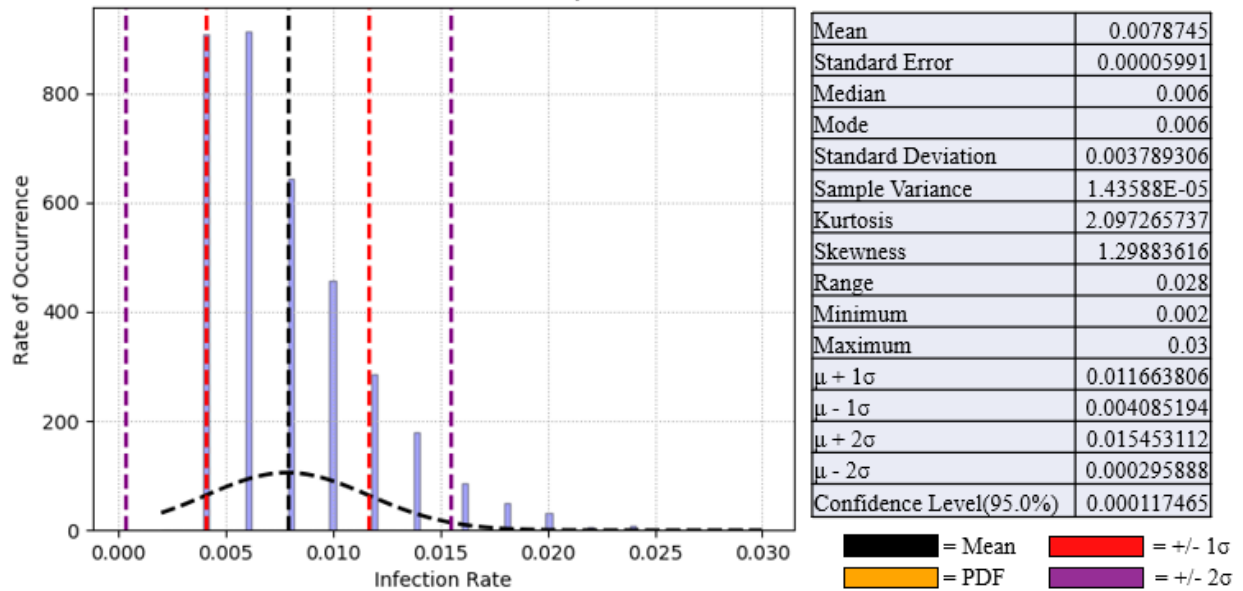
FEDERALWeb – Based : Data Set 4 (II) : $\mu = 51.304$, $\sigma = 6.541$



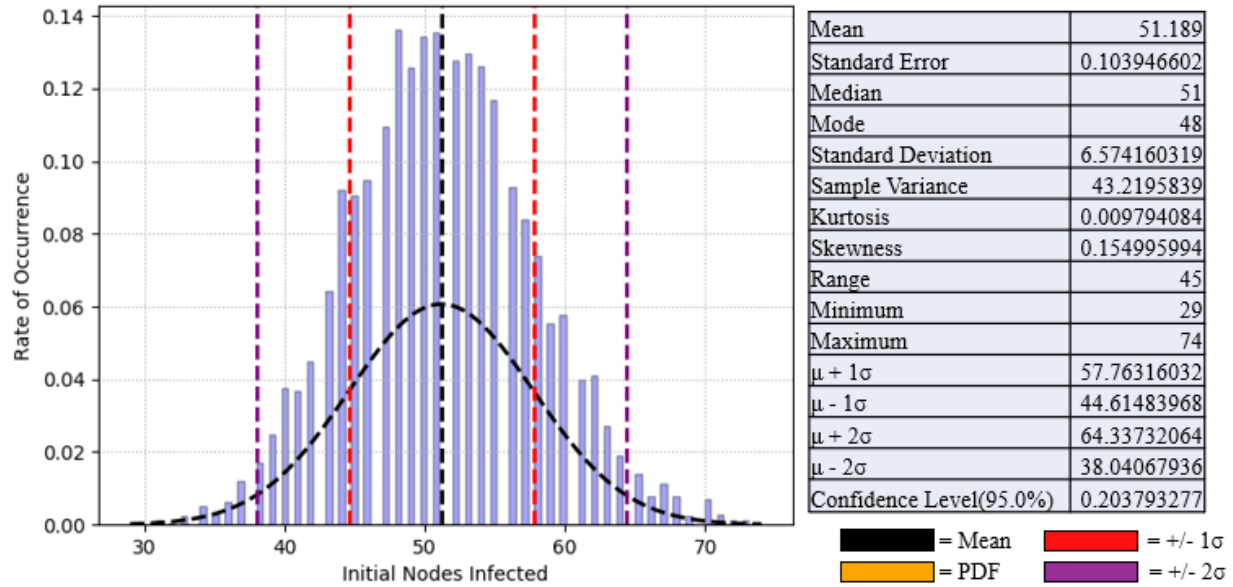
FEDERALWeb – Based : Data Set 4 (I@500) : $\mu = 55.242$, $\sigma = 6.85$



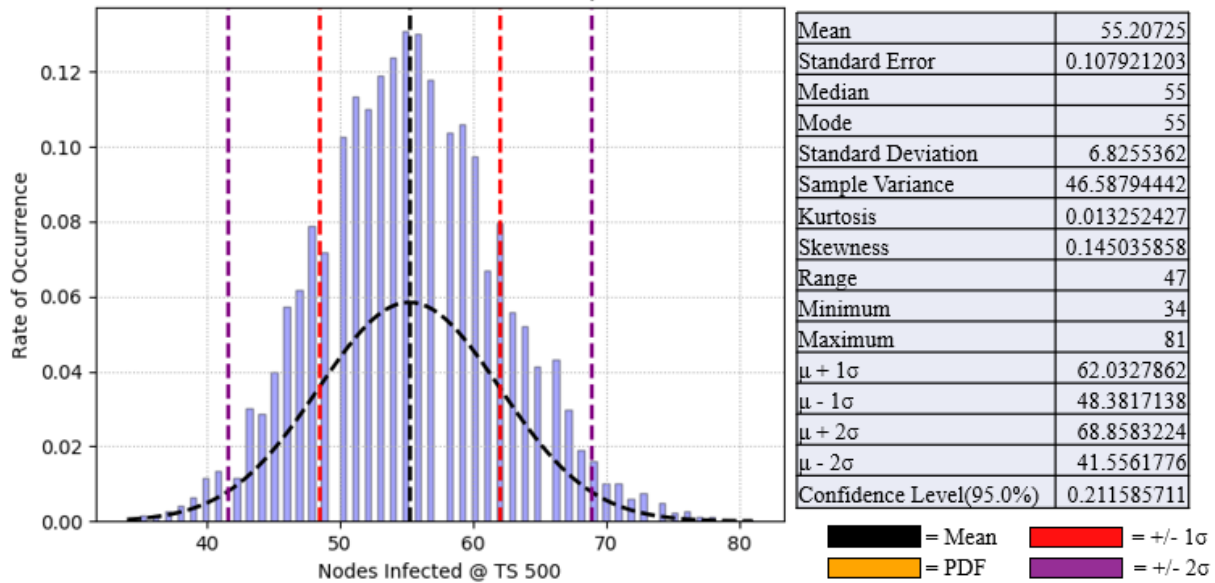
FEDERALWeb – Based : Data Set 4 (IR) : $\mu = 0.008$, $\sigma = 0.004$



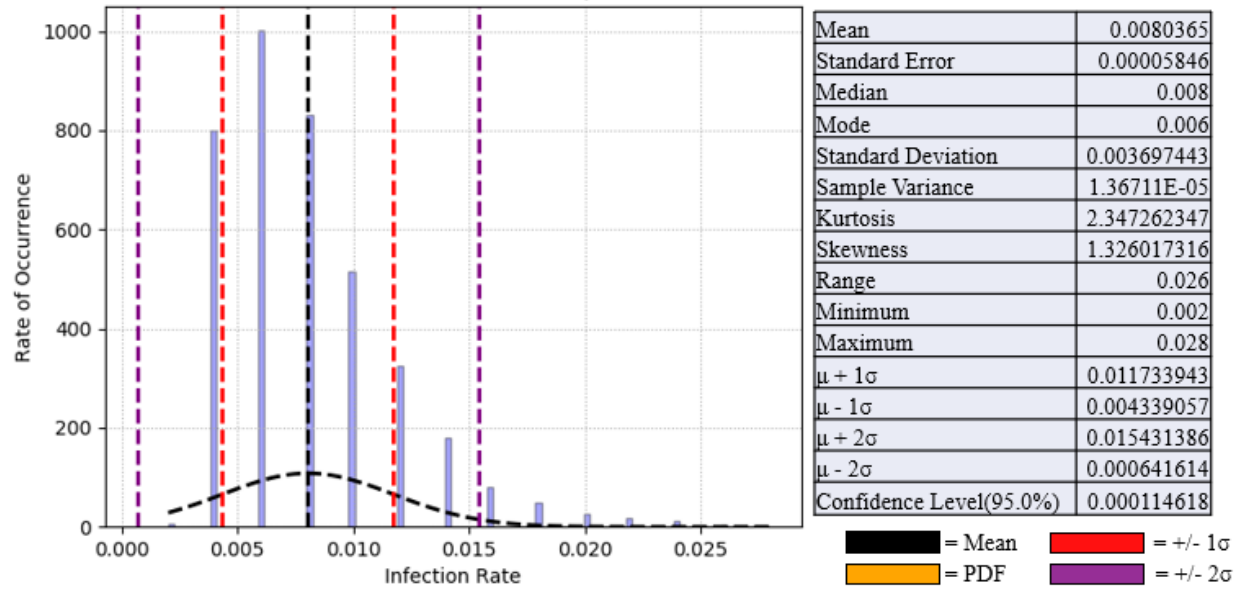
FEDERALWeb – Based : Data Set 5 (II) : $\mu = 51.189$, $\sigma = 6.573$



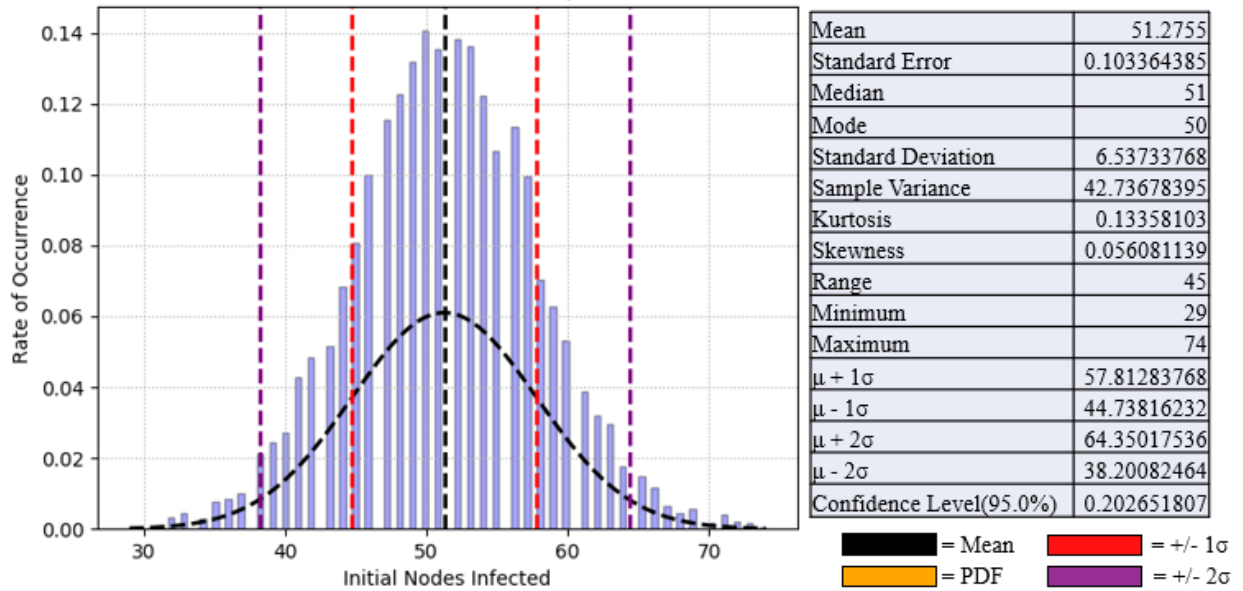
FEDERALWeb – Based : Data Set 5 (I@500) : $\mu = 55.207$, $\sigma = 6.821$



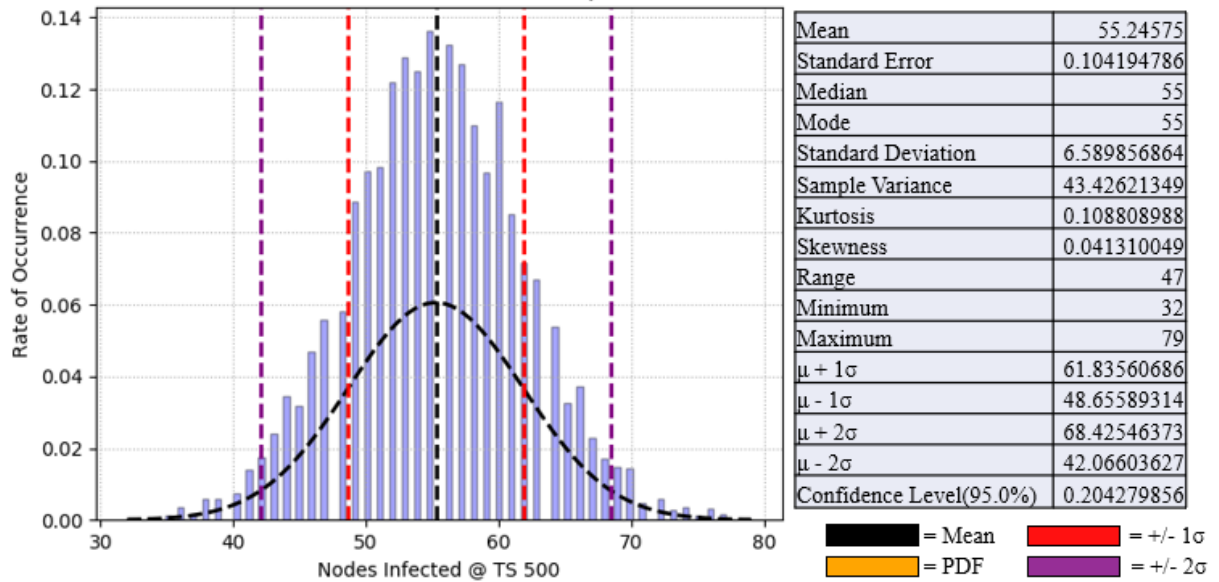
FEDERALWeb – Based : Data Set 5 (IR) : $\mu = 0.008$, $\sigma = 0.004$



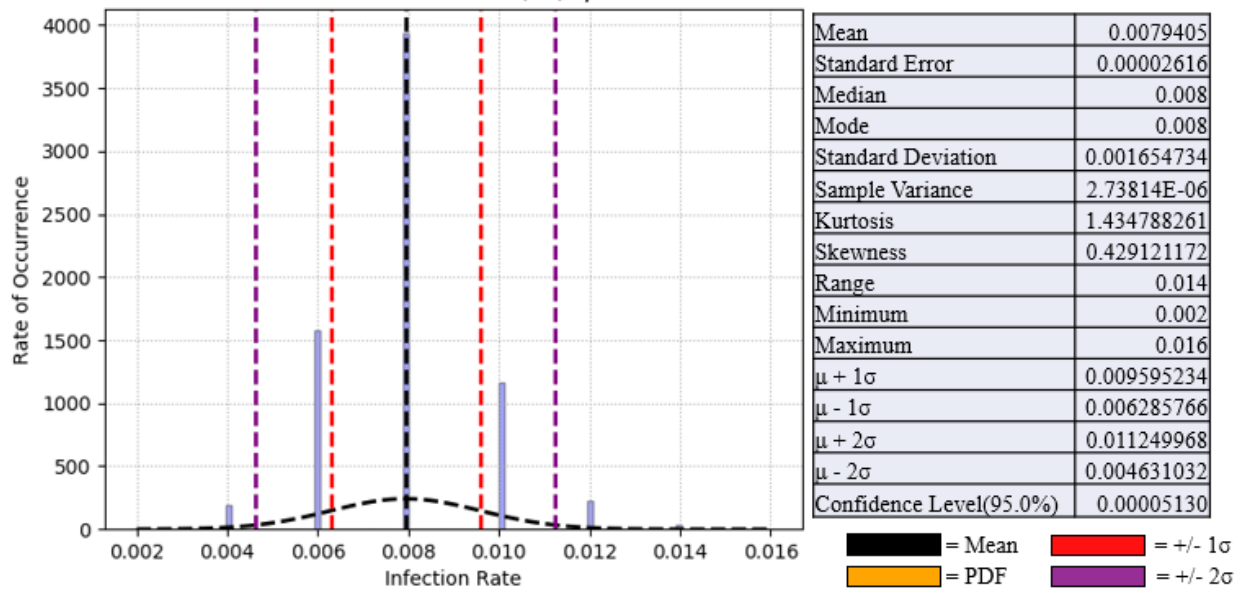
FEDERALWeb – Based : Data Set 6 (II) : $\mu = 51.276$, $\sigma = 6.537$



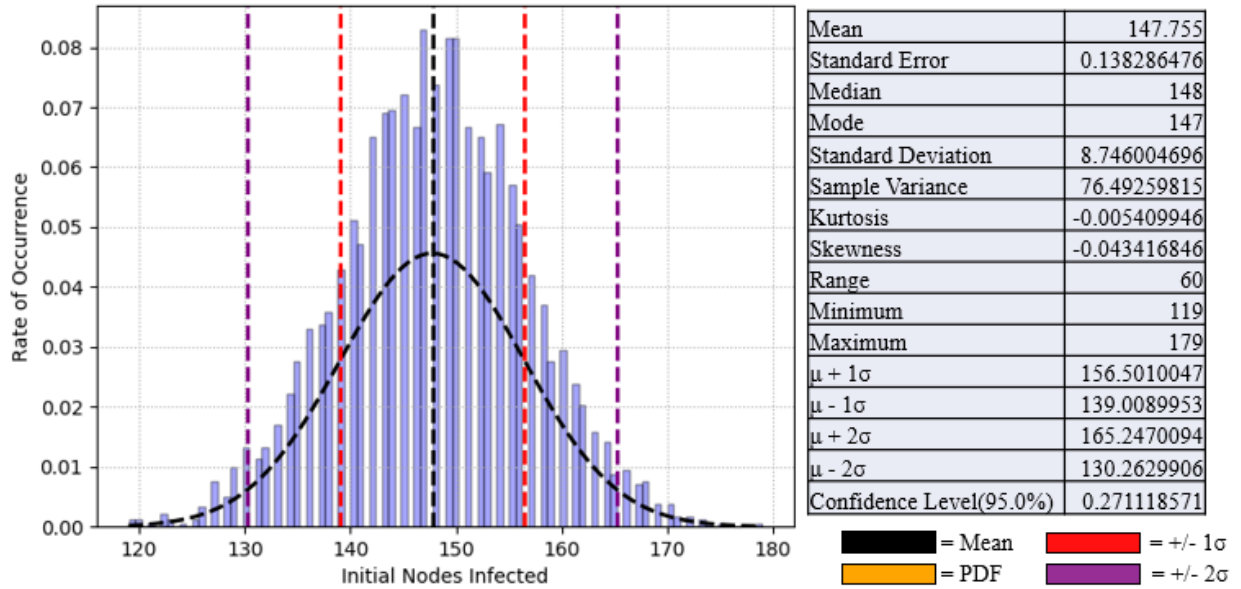
FEDERALWeb – Based : Data Set 6 (I@500) : $\mu = 55.246$, $\sigma = 6.58$



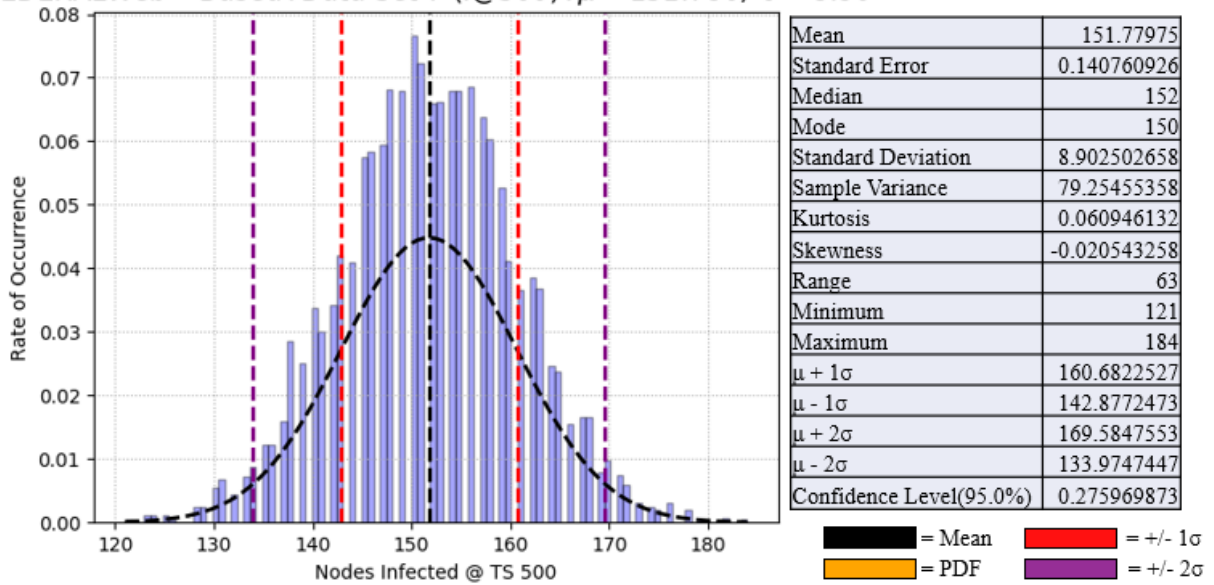
FEDERALWeb – Based : Data Set 6 (IR) : $\mu = 0.008$, $\sigma = 0.002$



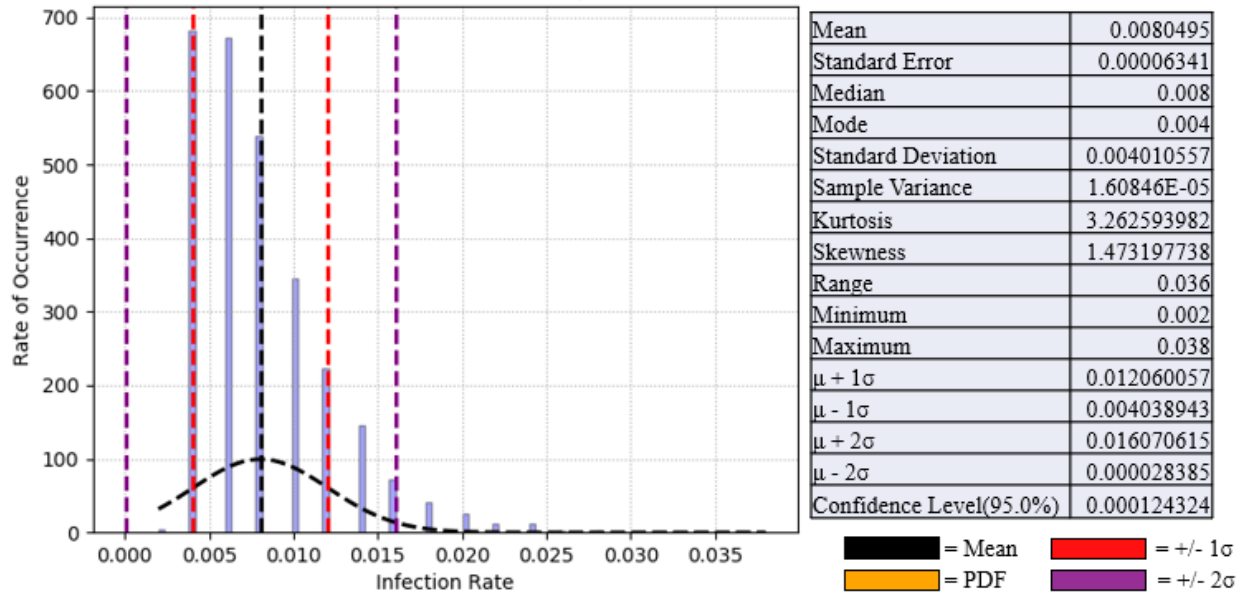
FEDERALWeb – Based : Data Set 7 (II) : $\mu = 147.755$, $\sigma = 8.745$



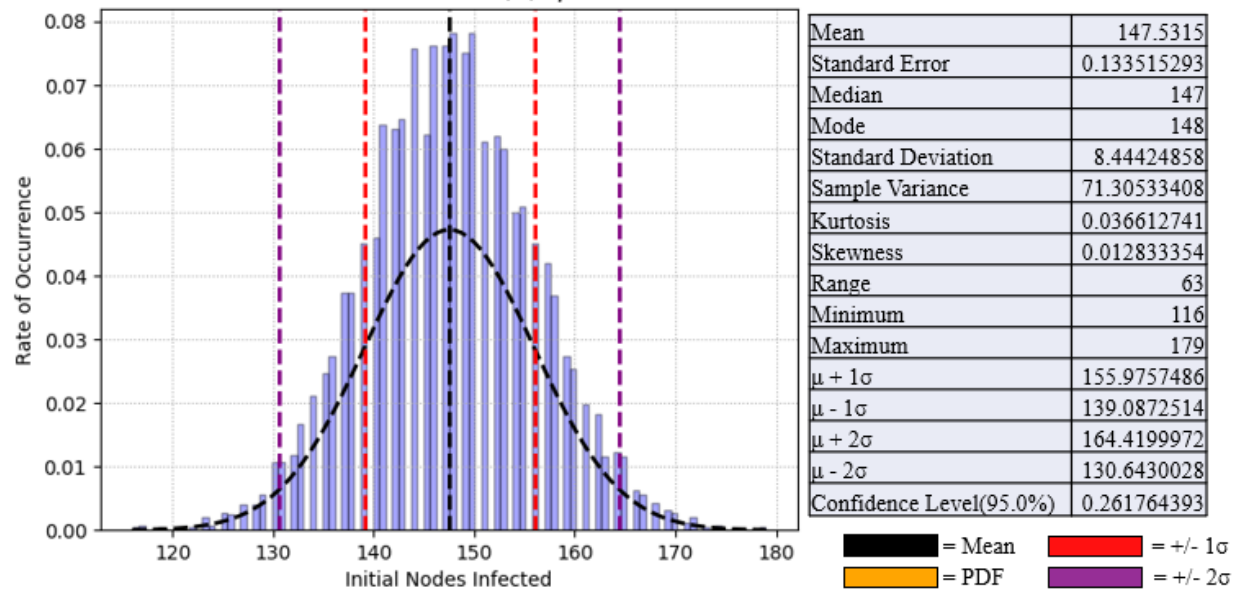
FEDERALWeb – Based : Data Set 7 (I@500) : $\mu = 151.780$, $\sigma = 8.90$



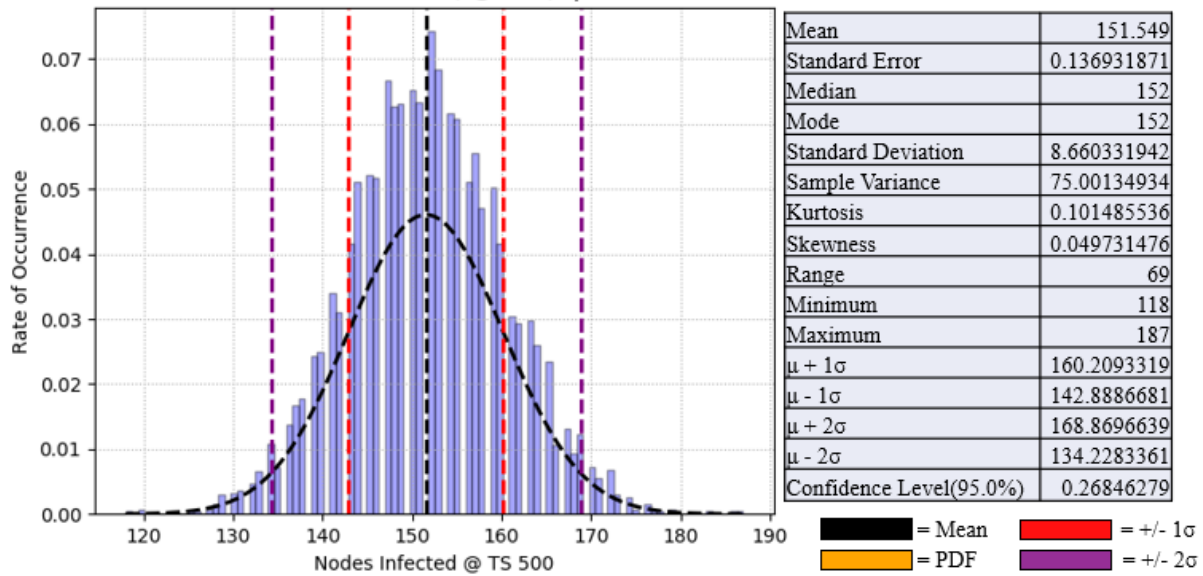
FEDERALWeb – Based : Data Set 7 (IR) : $\mu = 0.008$, $\sigma = 0.004$



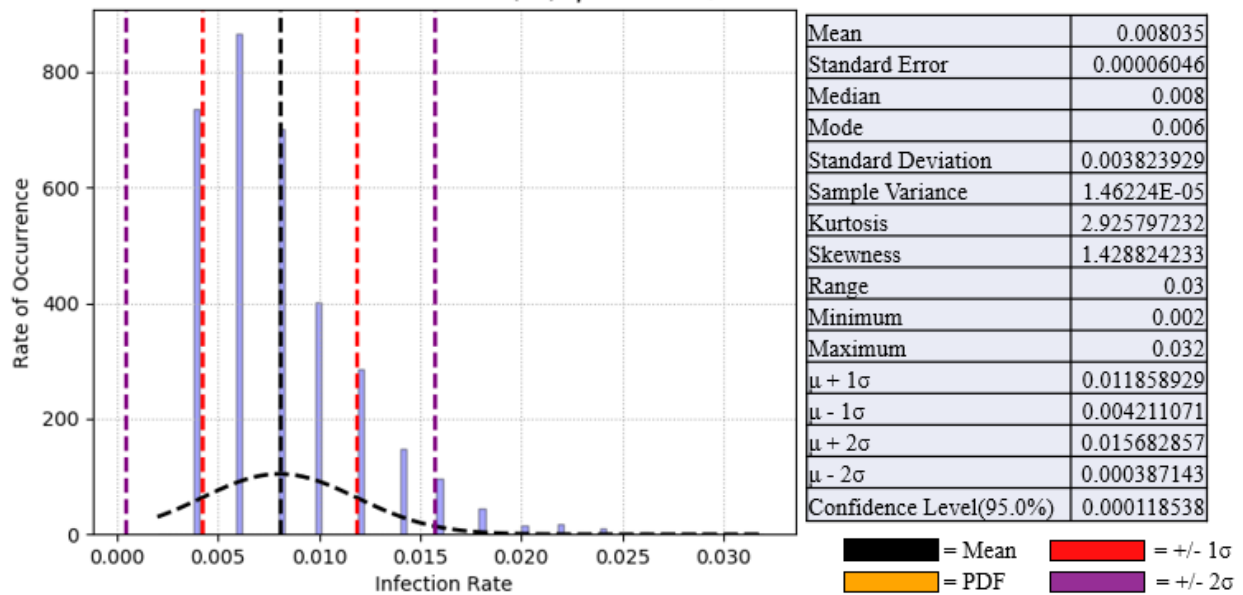
FEDERALWeb – Based : Data Set 8 (II) : $\mu = 147.531$, $\sigma = 8.443$



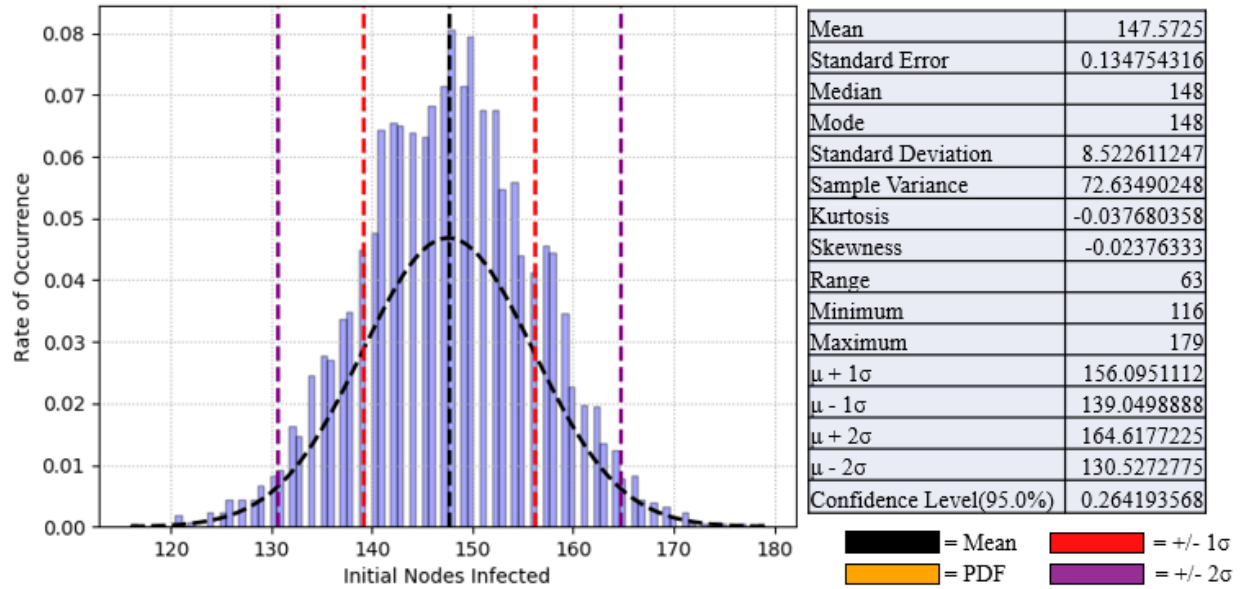
FEDERALWeb – Based : Data Set 8 (I@500) : $\mu = 151.549$, $\sigma = 8.65$



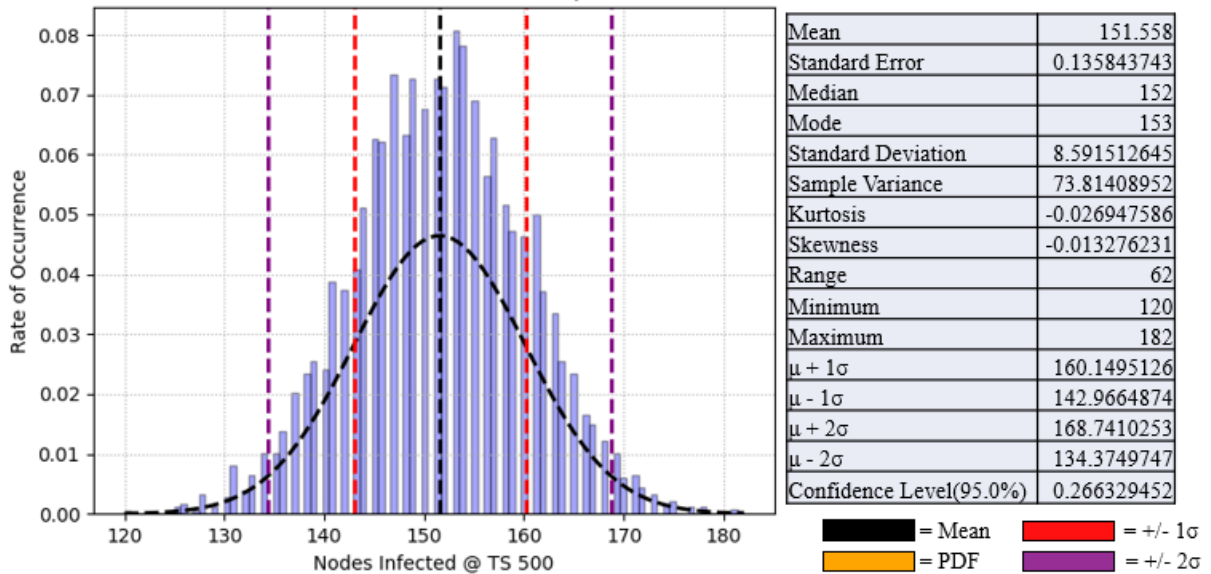
FEDERALWeb – Based : Data Set 8 (IR) : $\mu = 0.008$, $\sigma = 0.004$



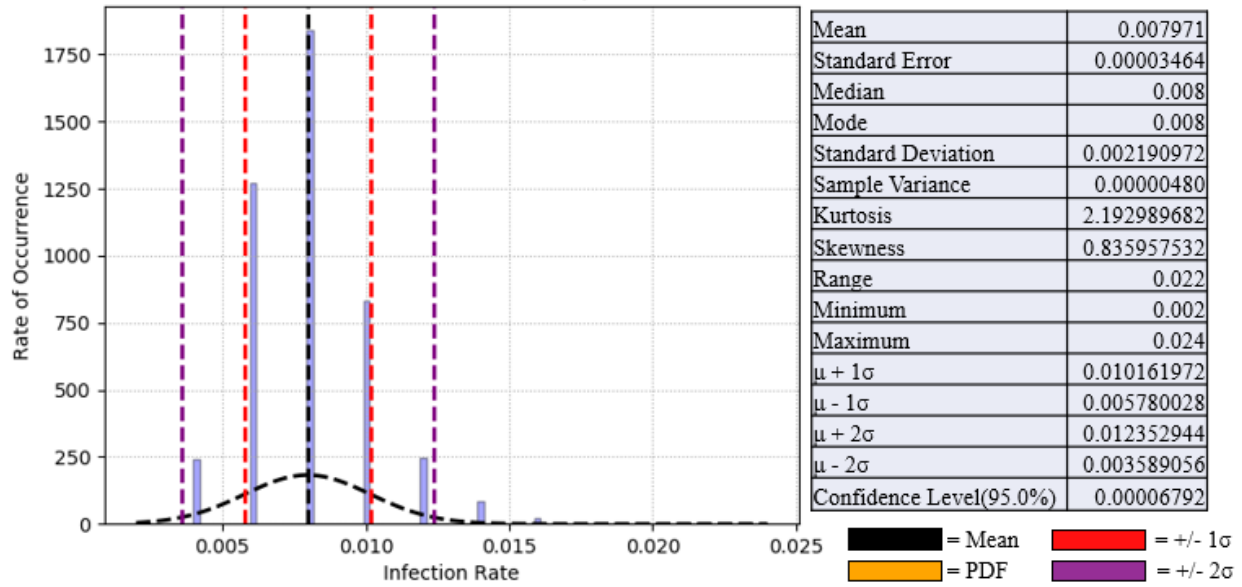
FEDERALWeb – Based : Data Set 9 (II) : $\mu = 147.572$, $\sigma = 8.522$



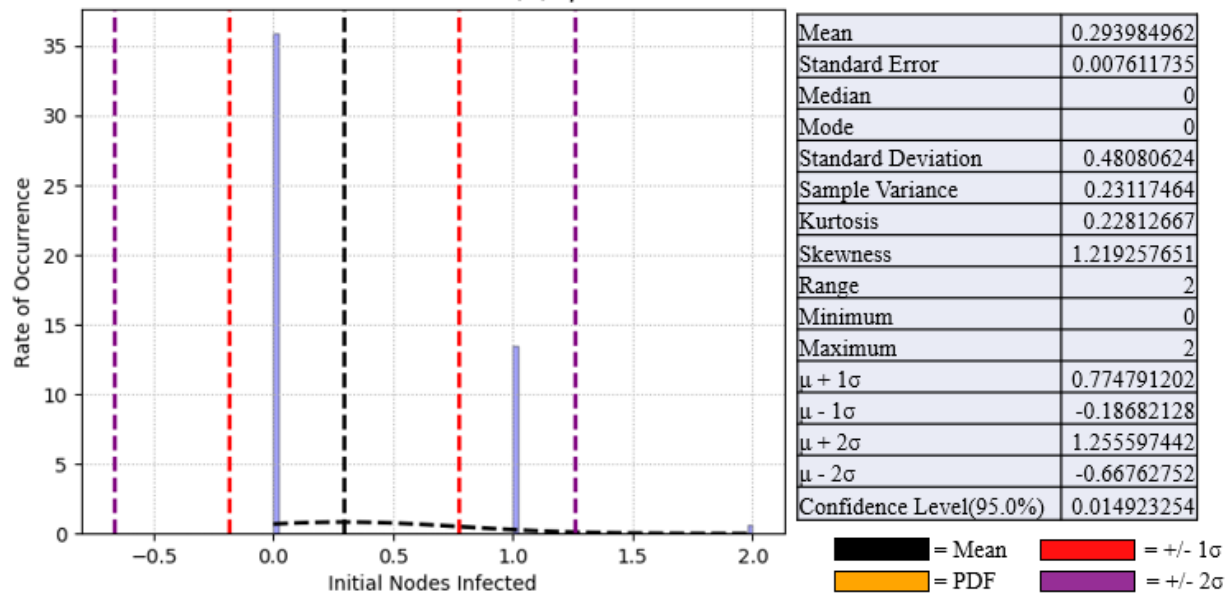
FEDERALWeb – Based : Data Set 9 (I@500) : $\mu = 151.558$, $\sigma = 8.59$



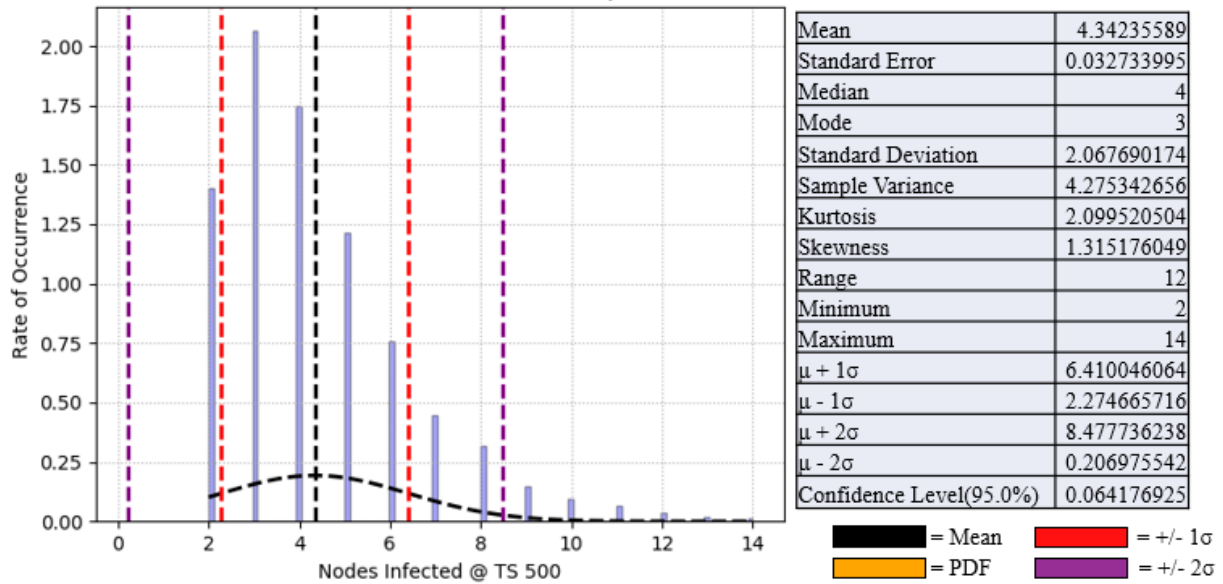
FEDERALWeb – Based : Data Set 9 (IR) : $\mu = 0.008$, $\sigma = 0.002$



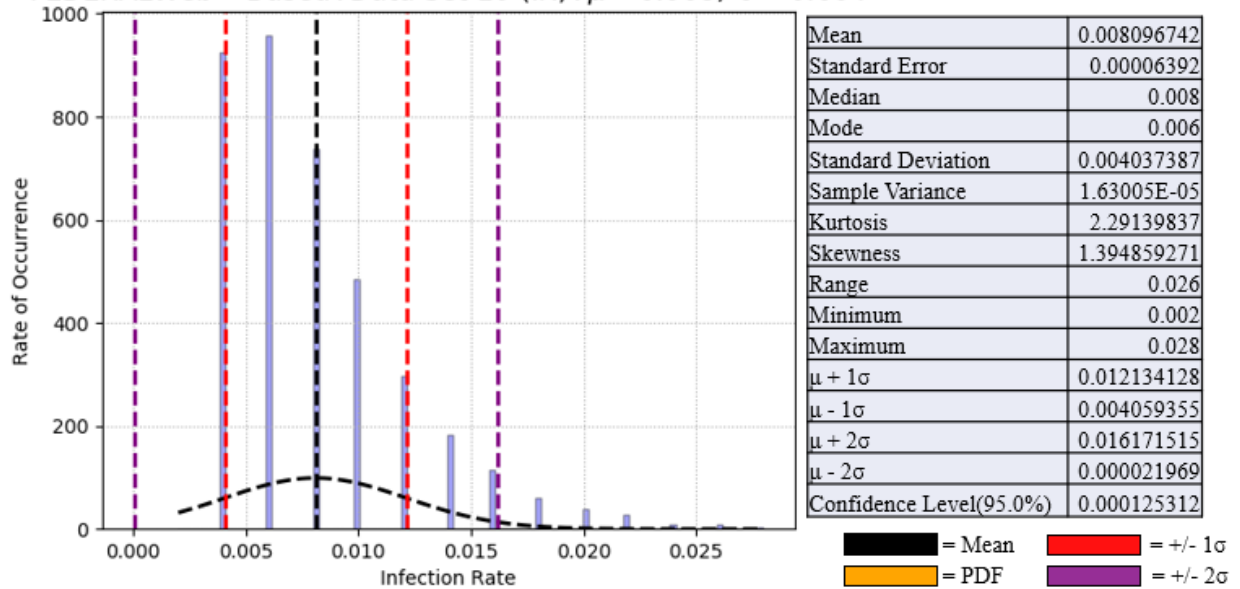
FEDERALWeb – Based : Data Set 10 (II) : $\mu = 0.294$, $\sigma = 0.481$



FEDERALWeb – Based : Data Set 10 (I@500) : $\mu = 4.342$, $\sigma = 2.06$



FEDERALWeb – Based : Data Set 10 (IR) : $\mu = 0.008$, $\sigma = 0.004$



REFERENCES

- Albert, R. J. (2000). Error tolerance and attack in complex networks. *Nature*, 406, 387-482.
- Anderson, R. &. (2006). The economics of information security. *Science*, 314(5799), 610-613.
- Burruss, G. W. (2010). Threatened globally, acting locally: Modeling law enforcement homeland security practices. *Justice Quarterly*, 27(1), 77-101.
- Caulkins, B. (2018, July 10). Behavioral Aspects of Cyber Security "Situation Awareness". Orlando, Florida, United states.
- Chawdry, R. (2017). *Accelerating the Data Sharing Process*. SypherLink.
- Cheng, Y., Deng, J., Li, J., Deloach, S., & Singhai, A. (2012). *Metrics of Security*. Retrieved from National Institute of Standards and Technology:
https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=917850
- Cohen, F. (1999). Simulating Cyber Attacks, Defences, and Consequences. *Computers and Security*, 18(6), 479-518.
- Department, P. J. (2018, June 25). *JNET Pennsylvania Justice Network* . Retrieved from Pennsylvania Government Website: Pennsylvania
- Department, P. J. (2018, August 28). Pennsylvania Justice Network. Pennsylvania Justice Network, United states. Retrieved from Pennsylvania Justice Network:
<https://www.pajnet.pa.gov>
- Dhillon, J. S. (2001). Defensive Information Operations and Domestic Law: Limitations on Government Investigative Techniques. *AFL Rev.*, 50, 135.
- Dr. Caulkins, B. (n.d.). M & S Behavioral CyberSecurity. Orlando, Florida, United states.
- Dutta, A. &. (2002). Management's role in information security in a cyber economy. *California Management Review*, 45(1), 67-87.
- Elkin-Koren, N. &. (1999). Law and economics in cyberspace. *International Review of Law and Economics*, 19(4), 553-581.
- ENFORCEMENT, F. D. (2017). *FLORIDA DEPARTMENT OF LAW ENFORCEMENT* .
FLORIDA DEPARTMENT OF LAW ENFORCEMENT .
- Enforcement, F. D. (2018, March 23). *Florida Department of Law Enforcement*. Retrieved from Florida Law Lenforcement: www.fdle.state.fl.us/
- FBI. (2018, August 23). *FBI.GOV*. Retrieved from FBI.GOV:
<https://www.fbi.gov/services/cjis/ndex>

- Fencel, T., Burget, P., & Bilek, J. (2011). Network Topology Design. *Control Engineering Practice*, Volume 19, Issue 11, Pages 1287 - 1296.
- Giandomenico, N. (2018, September 19). What is Spear-phishing? Defining and Differentiating Spear phishing from phishing. *Digital Guardian*. Retrieved from Digital Guardian.
- Gilbert, M., & Liebhold, A. (2010). Comparing methods for measuring the rate of spread of invading populations. *Ecography*, 33(5), 809-817.
- Greene, K. K., Kostick, J., Steves, M. P., & Theofanos, M. F. (2018). *User Context: An Explanatory Variable in Phishing Susceptibility*. San Diego, CA: National Institute of Standards and Technology.
- Groeneveld, R. A., & Meeden, G. (Dec., 1984). Measuring Skewness and Kurtosis. *Journal of the Royal Statistical Society. Series D (The Statistician)*, Vol. 33, No. 4, pp. 391-399 (9 pages).
- Hagberg, A., Schult, D., & Swart, P. (2018, September 20). NetworkX Reference. *NetworkX Reference Release 2.3rc1.dev20180920174416*. 2018, United states: Oxford University Press.
- Hawkins, D. (2013, July). Law Enforcement Guide for Communications Interoperability. *Office of Community Oriented Policing Services*. Washington D.C, United states: U.S. Department of Justice.
- Hinduja, S. (2004). Perceptions of local and state law enforcement concerning the role of computer crime investigative teams. *Policing: An International Journal of Police Strategies & Management*, 27(3), 341-357.
- Hong, J. W. (2011, July). An intrusion and defense testbed in a cyber-power system environment. *In Power and Energy Society General Meeting*, 2011 IEEE (pp. 1-5). IEEE.
- House, W. (2003). The national strategy to secure cyberspace. *Washington, DC: White House*.
- Jajodia, S. &. (2010). Advanced cyber attack modeling analysis and visualization. *GEORGE MASON UNIV FAIRFAX VA*.
- Kotenko, I. (2007, September). Multi-agent modelling and simulation of cyber-attacks and cyber-defense for homeland security. *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2007. IDAACS 2007*, 4th IEEE Workshop on (pp. 614-619).
- Kruger, H. A., & Kearney, W. D. (June 2006). A prototype for assessing information security. *Computers and Security*, vol. 25, no. 4, pp. 289-296.
- Kumar, V., Lazarevic, A., & Srivastava, J. (2005). *MANAGING CYBER THREATS Issues, Approches and Challenges*. New York: Springer Science and Business Media, Inc.

- Lala, C. &. (2001). Evaluating damage from cyber attacks: a model and analysis. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 31(4), 300-310.
- Liu, P. &. (2001, June). Multi-phase damage confinement in database systems for intrusion tolerance. *csfw*, p. 0191.
- Management, F. I. (2018). *FBI Services Information Management*. Retrieved from Privacy Impact Assessment for the National Data Exchange (N-DEx) System: <https://www.fbi.gov/services/information-management/foipa/privacy-impact-assessments/N-DEx>
- Mäses, S. (2015). *EVALUATION METHOD FOR HUMAN ASPECTS OF INFORMATION SECURITY*. Estonia: TALLINN UNIVERSITY OF TECHNOLOGY.
- Matulevicius, R. (2017). *Fundamentals of Secure System Modelling*. Cham, Switzerland: Springer International Publishing.
- Michael, K., & Michael, M. G. (2013). The future prospects of embedded microchips in humans as unique identifiers: the risks versus the rewards. *Media Culture and Society*, 78-86.
- Nagaraja, S. &. (2006). The Topology of Covert Conflict. *Paper presented at the Fifth Workshop on the Economics of Information Security, Cambridge*, weis2006.econinfosec.org/docs/38.pdf.
- Nascimento, C. d., & Mesquita, R. d. (2009). *A HUMAN ERROR PROBABILITY ESTIMATE METHODOLOGY BASED ON FUZZY INFERENCE AND EXPERT JUDGMENT ON NUCLEAR PLANTS*. Brazil: Centro Tecnológico da Marinha em São Paulo .
- NCIC, F. (2000, July 24). *National Crime Information Center (NCIC)*. Retrieved from FBI NCIC: National Crime Information Center (NCIC)
- NDEx, F. (2018, September 2). *National Data Exchange (N-DEx) System*. Retrieved from FBI Services: <https://www.fbi.gov/services/cjis/ndex>
- O'Brien, S. A. (2017). Giant Equifax Data Breach: 143 million people could be affected. *CNN Tech*.
- Scott, E. D. (2006). *FACTORS INFLUENCING USER-LEVEL SUCCESS IN POLICE INFORMATION SHARING: AN EXAMINATION OF FLORIDA'S FINDER SYSTEM* . Orlando, Florida, United states.
- Shulgin, B., Stone, L., & Agur, Z. (1998). Pulse vaccination strategy in the SIR epidemic model. *Bulletin of Mathematical Biology* , Volume 60, Issue 6, pp 1123 - 1148.

- Taylor, M. J., Epper, R. C., & Tolman, T. K. (1998). *state and local Law Enforcement Wireless Communications and Interoperability: A Quantitative Analysis*. Washington, DC: U.S. Department of Justice.
- Vázquez, D. F. (2012, June). Conceptual framework for cyber defense information sharing within trust relationships. *Cyber conflict (CYCON)*, 2012 4th international conference, pp. 1-17.
- Von Solms, R. &. (2013). From information security to cyber security. *computers & security*, 38, 97-102.
- Workman, M. (2008). Wisecrackers: A theory-grounded investigation of phishing and pretext social engineering threats to information security. *Journal of the American Society for Information Science and Technology*, 59(4), 662-674.