Journal of Information Security, 2016, 7, 112-128
Published Online April 2016 in SciRes. http://www.scirp.org/journal/jis
http://dx.doi.org/10.4236/jis.2016.73008

Cybersecurity: A Statistical Predictive Model for the Expected Path Length

Pubudu Kalpani Kaluarachchi*, Chris P. Tsokos, Sasith M. Rajasooriya

Department of Mathematics and Statistics, University Of South Florida, Tampa, FL, USA
Email: *pubudu@mail.usf.edu

Received 1 March 2016; accepted 2 April 2016; published 5 April 2016

Abstract

The object of this study is to propose a statistical model for predicting the Expected Path Length (expected number of steps the attacker will take, starting from the initial state to compromise the security goal—EPL) in a cyber-attack. The model we developed is based on utilizing vulnerability information along with having host centric attack graph. Utilizing the developed model, one can identify the interaction among the vulnerabilities and individual variables (risk factors) that drive the Expected Path Length. Gaining a better understanding of the relationship between vulnerabilities and their interactions can provide security administrators a better view and an understanding of their security status. In addition, we have also ranked the attributable variables and their contribution in estimating the subject length. Thus, one can utilize the ranking process to take precautions and actions to minimize Expected Path Length.

Keywords

Vulnerability, Attack Graph, Markov Model, Security Evaluation, Expected Path Length, CVSS

1. Introduction

Cyber-attacks are the most formidable security challenge faced by most governments and large scale companies. Cyber criminals are increasingly using sophisticated network and social engineering techniques to steal the crucial information which directly affects the operational effects of the Government or Company's objectives. According to the Secunia [1] report 2015, one could see how crucial the volume and magnitude of increasing cyber-security threaten. Thus, in understanding the performance, availability and reliability of computer networks, measuring techniques plays an important role in the subject area.

Quantitative measures are now commonly used to evaluate the security of computer network systems. These

*Corresponding author.

How to cite this paper: Kaluarachchi, P.K., Tsokos, C.P. and Rajasooriya, S.M. (2016) Cybersecurity: A Statistical Predictive Model for the Expected Path Length. *Journal of Information Security*, **7**, 112-128. http://dx.doi.org/10.4236/jis.2016.73008

measures help administrators to make important decisions regarding their network security.

In the present study, we have first proposed a stochastic model for security evaluation based on vulnerability exploitability scores and attack path behavior. Here, we consider small case scenarios which include three vulnerabilities (high, medium and small) as a base model to understand the behavior of network topology. We structure the attack graph which includes all possibilities that the attacker reach the goal state and use probabilistic analysis to measure the security of the network. In addition, we propose a statistical model that is driven by the mentioned vulnerabilities along with the significant interactions that is highly accurate. This statistical model will allow us to estimate the Expected Path Length and Minimum number of steps to reach the target with probability one. Having these important estimates, we can take counter steps and acquire relevant resources to protect the security system from the attacker. In addition, utilizing this model we have identified the significant interaction of the key attributable variables. Also we can rank the attributable variables (vulnerabilities) to identify the percentage of contribution to the response (Expected Path Length and Minimum number of steps to reach the target) and furthermore one can perform surface response analysis to identify the acceptable values that will minimize the Expected Path Length among others.

2. Background and Terms of Cybersecurity

Here we review some of the terminology associated with cyber security for the convenience of the reader. We also describe some basic aspects of Markov chains properties that we utilized in fulfilling the objectives of the present study.

Figure 1 and **Figure 2** below give a schematic presentation of the Common Vulnerability Scoring System (CVSS) which is the basis of the metric calculation model and the temporal and environmental matrices calculation model, respectively.

2.1.1. Vulnerabilities

In computer security, a vulnerability [2]-[4] is a weakness which allows an attacker to reduce a system's information assurance. Vulnerability is the intersection of three elements, which are, systems susceptibility to the

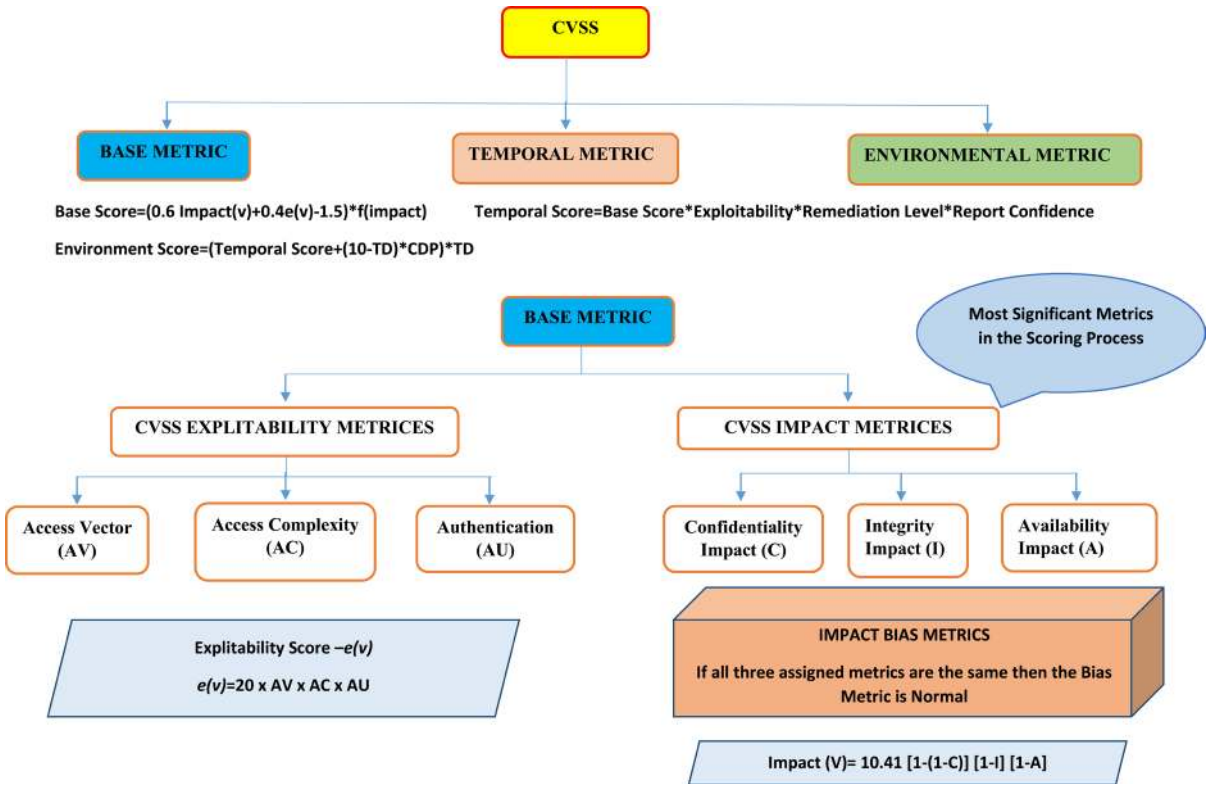

Figure 1. Common vulnerability scoring system-base metric calculation model.

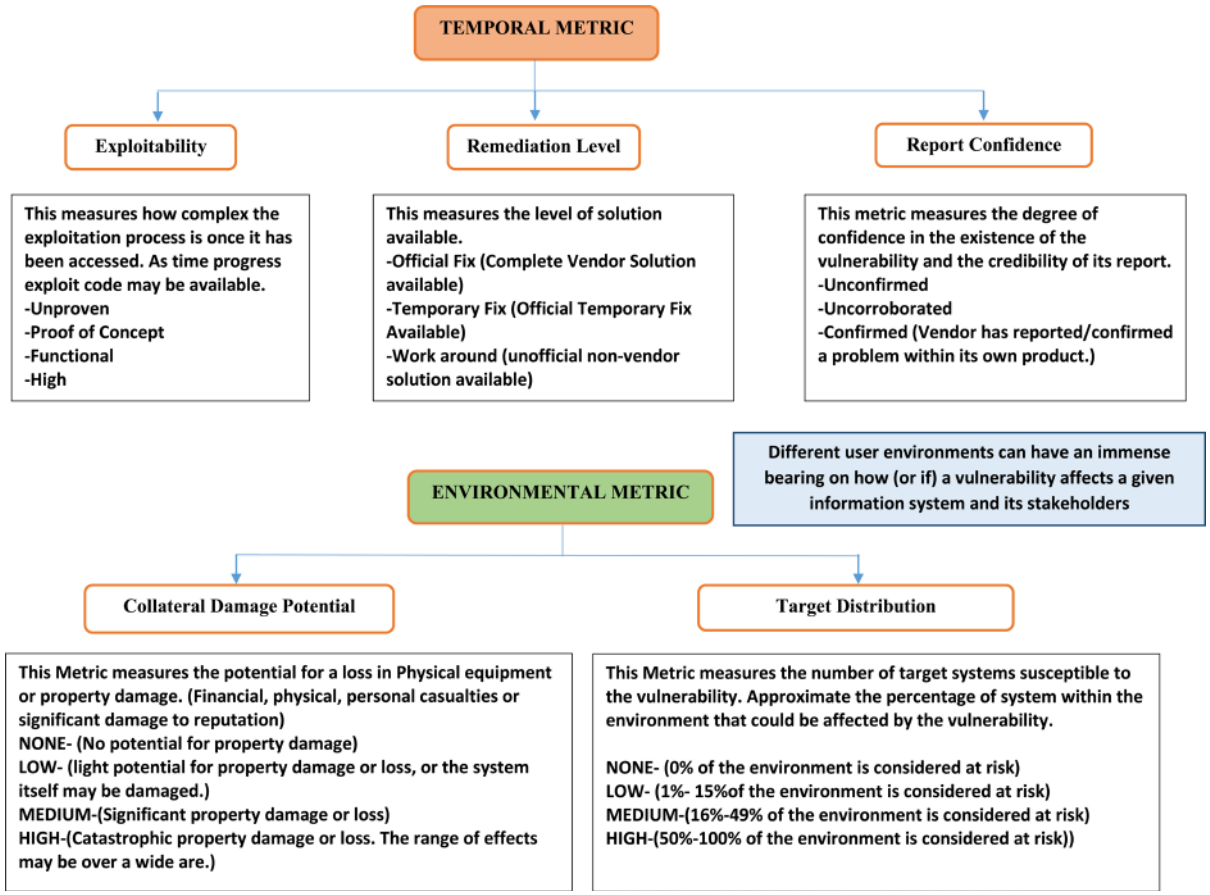

Figure 2. Common vulnerability scoring system-temporal and environmental metrics calculation model.

flaw, attacker access to the flaw, and attacker capability to exploit the flaw.

To exploit a vulnerability, an attacker must have at least one applicable tool or technique that can connect to a system weakness. In this frame, vulnerability is also known as the attack surface.

The attack surface of a software environment is the sum of the different points (the "attack vectors") where an unauthorized user (the "attacker") can try to enter data to or extract data from an environment.

2.1.2. Attack Graphs

An attack graph [5] [6] is a succinct representation of all paths through a system that ends in a state where an intruder has successfully achieved his goal.

Attack graphs describe ways in which an adversary can exploit vulnerabilities to break into a system. System administrators analyze attack graphs to understand where their system's weaknesses lie and to help decide which security measures will be effective to deploy. In practice, attack graphs are produced manually by Red Teams. Construction by hand, however, is tedious, error-prone, and impractical for attack graphs with large number of nodes.

2.1.3. Frei's Vulnerabilities Lifecycle

Frei's Vulnerability Lifecycle [7] is a representation of stages that vulnerability faces with time. This model calculates the likelihood of an exploit or patch being available a certain number of days after its disclosure date.

2.1.4. Common Vulnerability Scoring System (CVSS)

Common Vulnerability Scoring System (CVSS) [8] is a free and open industry standard for assessing the severity of computer system security vulnerabilities. It is under the custodianship of the Forum of Incident Response and Security Teams (FIRST). It attempts to establish a measure of how much concern a vulnerability warrants,

compared to other vulnerabilities, so efforts can be prioritized. The scores are based on a series of measurements (called metrics) based on expert assessment. The scores range from 0 to 10. Vulnerabilities with a base score in the range 7.0 - 10.0 are High, those in the range 4.0 - 6.9 as Medium, and 0 - 3.9 as Low. CVSS calculating method is described by **Figure 1** and **Figure 2** are given in Section 2.

2.1.5. Cyber Situational Awareness

Tim Bass [9] first introduced this concept and this is the immediate knowledge of friendly, adversary and other relevant information regarding activities in and through cyberspace and the Electromagnetic Spectrum (EMS). It is obtained from a combination of intelligence and operational activity in cyberspace, the EMS, and in the other domains, both unilaterally and through collaboration with our unified action and public-private partners.

Cyber situational awareness is the capability that helps security analysts and decision makers:

- Visualize and understand the current state of the IT infrastructure, as well as the defensive posture of the IT environment.
- Identify what infrastructure components are important to complete key functions.
- Understand the possible actions an adversary could undertake to damage critical IT infrastructure components.
- Determine where to look for key indicators of malicious activity.

2.2. Markov Chain and Transition Probability

A discrete type stochastic process $X=\{X_N, N\geq 0\}$ is called a Markov chain [10] if for any sequence $\{X_0, X_1, \cdots, X_N\}$ of states, the next state depends only on the current state and not on the sequence of events that preceded it, which is called the Markov property. Mathematically we can write this as follows:

$$P\left(X_N=j \middle| X_0=i_0, X_1=i_1, \cdots, X_{N-2}=i_{N-2}, X_{N-1}=i\right)=P\left(X_N=j \middle| X_{N-1}=i\right).$$

We will also make the assumption that the transition probabilities $P\left(X_N=j \middle| X_0=i_0, X_1=i_1, \cdots, X_{N-2}=i_{N-2}, X_{N-1}=i\right)$ do not depend on time. This is called time homogeneity. The transition probabilities ($P_{i,j}$) for Markov chain can be defined as follows:

$$P_{i,j}=P\left(X_N=j \middle| X_{N-1}=i\right).$$

The transition matrix P of the Markov chain is the $N\times N$ matrix whose (i,j) entry $P_{i,j}$ satisfied the following properties.

$$0\leq P_{ij}\leq 1, \quad 1\leq i,j\leq N$$

and

$$\sum_{j=1}^{N} P_{ij}=1, \quad 1\leq i\leq N.$$

Any matrix satisfying the above two equations is the transition matrix for a Markov chain.

To simulate a Markov chain, we need its stochastic matrix $\boldsymbol{P}$ and an initial probability distribution π_o.

Here we shall simulate an N-state Markov chain (X; P; π_0) for $N=0,1,2,\cdots,N$, time periods. Let X be a vector of possible state values from sample realizations of the chain. Iterating on the Markov chain will produce a sample path $\{X_N\}$ where for each N, $X_N \in X$. When writing simulation programs this is about using uniformly distributed U [0, 1] random numbers to obtain the corrected distribution in every step.

Transient States

Let P be the transition matrix [10] for Markov chain X_n. A "state i" is called transient state if with probability 1 the chain visits i only a finite number of times. Let Q be the sub matrix of P which includes only the rows and columns for the transient states. The transition matrix for an absorbing Markov chain has the following canonical form.

$$P=\begin{pmatrix} Q & R \\ 0 & I \end{pmatrix}.$$

Here P is the transition matrix, Q is the matrix of transient states, R is the matrix of absorbing states and I is the identity matrix.

The matrix P represents the transition probability matrix of the absorbing Markov chain. In an absorbing Markov chain the probability that the chain will be absorbed is always 1. Hence, we have

$$Q^n \to 0 \quad \text{as } n \to \infty .$$

Thus, is it implies that all the eigenvalues of Q have absolute values strictly less than 1. Hence, $I - Q$ is an invertible matrix and there is no problem in defining the matrix

$$M = \left(I - Q\right)^{-1} = I + Q + Q^2 + Q^3 + \cdots .$$

This matrix is called the fundamental matrix of P. Let i be a transient state and consider Y_i, the total number of visits to i. Then we can show that the expected number of visits to i starting at j is given by M_{ij}, the (i, j) entry of the matrix M.

Therefore, if we want to compute the expected number of steps until the chain enters a recurrent class, assuming starting at state j, we need only sum M_{ij} over all transient states i.

3. Cybersecurity Analysis Method

The core component of this method is the attack graph [11]. When we draw an attack graph for a cybersecurity system it has several nodes which represent the vulnerabilities that the system has and the attacker's state [12]. We consider that it is possible to go to a goal state starting from any other state in the attack graph. Also an attack graph has at least one absorbing state or goal state. Therefore we will model the attack graph as an absorbing Markov chain [12].

Absorbing state or goal state is the security node which is exploited by the attacker. When the attacker has reached this goal state, attack path is completed. Thus, the entire attack graph consists of these type of attack paths.

Given the CVSS score for each of the vulnerabilities in the attack Graph, we can estimate the transition probabilities of the absorbing Markov chain by normalizing the CVSS scores over all the edges starting from the attacker's source state.

We define,

p_{ij} = probability that an attacker is currently in state j and exploits a vulnerability in state i.

n = number of outgoing edges from state i in the attack model.

v_j = CVSS score for the vulnerability in state j.

Then formally we can define the transition probability below,

$$p_{ij} = \frac{v_j}{\sum_{k=1}^{n} v_k} .$$

By using these transition probabilities we can derive the absorbing transition probability matrix P, which follows the properties defined under Markov chain probability method.

3.1. Attack Prediction

Under the Attack prediction, we consider two methods to predict the attacker's behavior.

3.1.1. Multi Step Attack Prediction

The absorbing transition probability matrix shows the presence of each edge in a network attack graph. This matrix shows every possible single-step attack. In other words, the absorbing transition probability matrix shows attacker reaches ability within one attack step. We can navigate the absorbing transition probability matrix by iteratively matching rows and columns to follow multiple attack steps, and also raise the absorbing transition probability matrix to higher powers, which shows multi-step attacker reach ability at a glance.

For a square ($n \times n$) adjacency matrix P and a positive integer k, then P^k is P raised to the power k: Since P is an absorbing transition probability matrix with time, this matrix goes to some stationary matrix Π, where the rows of this matrix are identical. That is,

$$\lim_{k\to\infty} P^k = \Pi.$$

At the goal state column of this matrixΠ has ones, so we can find the minimum number of steps that the attacker should try to reach to the goal state with probability 1.

3.1.2. Prediction of Expected Path Length (EPL)

The Expected Path Length (EPL) measures the expected number of steps the attacker will take starting from the initial state to reach the goal state (the attacker's objective). As we discussed earlier P has the following canonical form.

$$P = \begin{pmatrix} Q & R \\ 0 & I \end{pmatrix}.$$

Here, P is the transition matrix, Q is the matrix of transient states, R is the matrix of absorbing states and I is the identity matrix.

The matrix P represents the transition probability matrix of the absorbing Markov chain. In an absorbing Markov chain the probability that the chain will be absorbed is always 1. Thus, we have

$$Q^n \to 0 \quad \text{as } n \to \infty .$$

This implies that all the eigenvalues of Q have absolute values strictly less than 1. Thus, $I - Q$ is an invertible matrix and there is no problem in defining the matrix

$$M = (I - Q)^{-1} = I + Q + Q^2 + Q^3 + \cdots .$$

Using this fundamental matrix M of the absorbing Markov chain we can compute the expected total number of steps to reach the goal state until absorption.

Taking the summation of first row elements of matrix M gives the expected total number of steps to reach the goal state until absorption and the probability value relates to the goal state gives the expected number of visits to that state before absorption.

4. Illustration: The Attacker

To illustrate the proposed approach model that we discussed in section 3, we considered a Network Topology [3] [12]-[14] given in **Figure 3**, below.

The network consists of two service hosts IP 1, IP 2 and an attacker's workstation, Attacker connecting to each of the servers via a central router.

In the server IP 1 the vulnerability is labeled as CVE 2006-5794 and let's consider this as V_1.

In the server IP 2 there are two recognized vulnerabilities, which are labeled CVE 2004-0148 and CVE 2006-5051. Let's consider this as V_2 and V_3, respectively.

We proceed to use the CVSS score of the above vulnerabilities. And the exploitability score (e (v) in **Figure 1**) of each vulnerabilities as given in **Table 1**, below.

Host Centric Attack graph

The host centric attack graph is shown by **Figure 4**, below. Here, we consider that the attacker can reach the

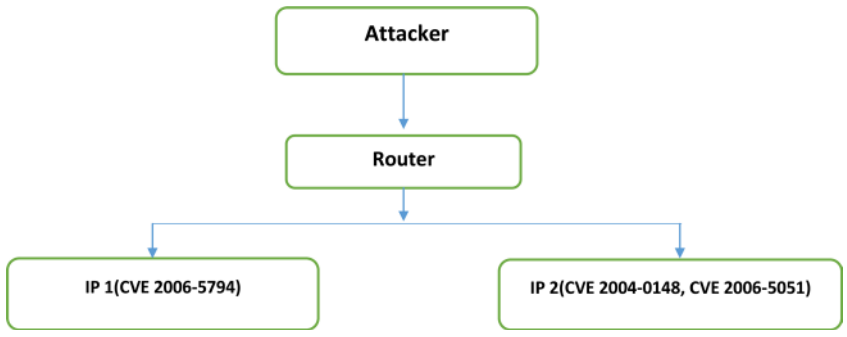

Figure 3. Network topology.

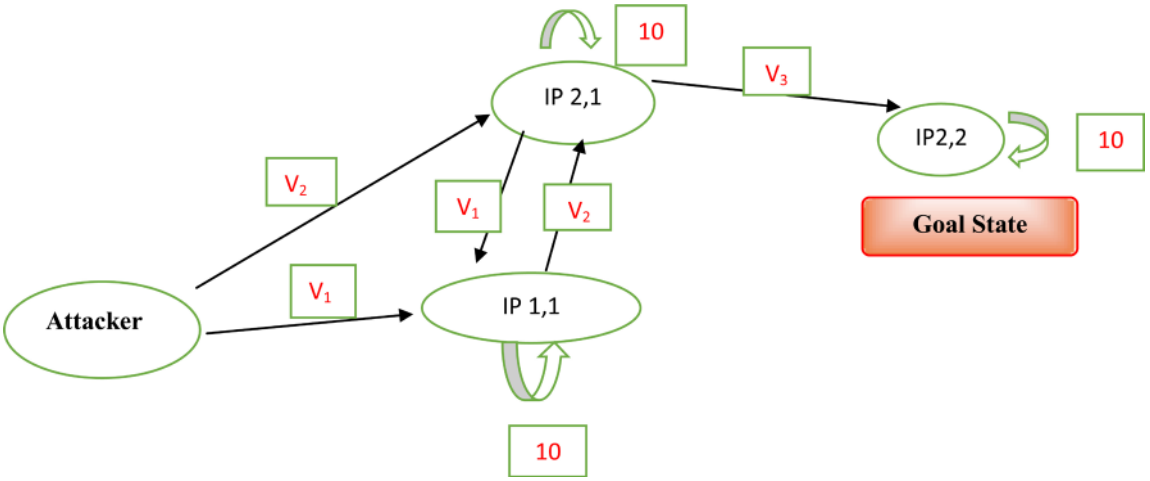

Figure 4. Host centric attack graph.

goal state only by exploiting V_2 vulnerability. The graph shows all the possible paths that the attacker can follow to reach the goal state.

Note that IP1,1 state represents V_1 vulnerability and IP2,1 and IP2,2 states represent V_2 and V_3 vulnerabilities, respectively. Also, the notation "10" represents the maximum vulnerability score and this provides attacker the maximum chance to exploit this state. Attacker can reach each state by exploiting the relevant vulnerability.

4.1. Adjacency Matrix for the Attack Graph

Let s_1, s_2, s_3, s_4, represent the attack states for Attacker, (IP1,1), (IP2,1) and (IP2,2), respectively.

To find the weighted value of exploiting each vulnerability from one state to another state, we divide the vulnerability score by summation of all out going vulnerability values from that state.

For our attack graph the weighted value of exploiting each vulnerability is given below.

1st row probabilities:

Weighted value of exploiting V_1 from s_1 to s_2 is $V_1/(V_1+V_2)$.

Weighted value of exploiting V_2 from s_1 to s_3 is $V_2/(V_1+V_2)$.

2nd row probabilities:

Weighted value of exploiting V_2 from s_2 to s_3 is $V_2/(10+V_2)$.

3rd row probabilities:

Weighted value of exploiting V_1 from s_3 to s_2 is $V_1/(V_1+V_3+10)$.

Weighted value of exploiting V_3 from s_3 to s_4 is $V_3/(V_1+V_3+10)$.

4th row probabilities:

Weighted value of exploiting V_3 from s_4 to s_4 is 1.

For the Host Centric Attack graph we can have the Adjacency Matrix as follows.

$$A = \begin{array}{c} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \end{array} \begin{array}{c} \begin{array}{cccc} s_1 & s_2 & s_3 & s_4 \end{array} \\ \begin{bmatrix} \frac{0}{V_1+V_2} & \frac{V_1}{V_1+V_2} & \frac{V_2}{V_1+V_2} & \frac{0}{V_1+V_2} \\ \frac{0}{10+V_2} & \frac{10}{10+V_2} & \frac{V_2}{10+V_2} & \frac{0}{10+V_2} \\ \frac{0}{V_1+V_3+10} & \frac{V_1}{V_1+V_3+10} & \frac{10}{V_1+V_3+10} & \frac{V_3}{V_1+V_3+10} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

Utilizing the information given in **Table 1**, the matrix A is given by

$$A = \begin{bmatrix} 0 & 0.5455 & 0.4545 & 0 \\ 0 & 0.6667 & 0.3333 & 0 \\ 0 & 0.3529 & 0.5882 & 0.0588 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Here, 0.5455 is the probability that attacker exploit V_1 vulnerability in first step, from s_1 to s_2. We can explain 0.0588 as the probability that once in state IP2,1 can exploit V_3 vulnerability and reach to IP2,2 in first attempt. Similarly each probability represents the chance to exploit relevant vulnerability from one state in the first attempt.

We want to use this matrix to answer the important question in cyber security analysis. We want to find the minimum number of steps to reach the goal state (final destination) with probability one and the expected path length metric.

4.2. Finding Stationary Distribution and Minimum Number of Steps

By using the above matrix A, we can find the probabilities with two, three and several attempt by the attacker to reach the goal state using $A^2, A^3, A^4, \cdots, A^p$ matrices. From these matrices we can find all possible probabilities from one state to another that the attacker can reach by two steps A^2, three steps A^3 and four steps A^4 and up to p steps A^p respectively. We continuous this process until we reach the absorbing matrix and that p value gives the minimum number of steps that the attacker is required to reach the goal state with probability one.

We proceed by changing the CVSS score and calculate for each combination of V_1, V_2 and V_3 the minimum number of steps that the attacker will reach the goal state with probability one. These calculations are given in **Table 2**, below.

Table 1. Vulnerability scores.

Vulnerability	Exploitability score
V_1 (CVE 2006-5794)	6
V_2 (CVE 2006-5051)	5
V_3 (CVE 2004-0148)	1

Table 2. Number of steps for absorbing matrix.

# of steps	V_1	V_2	V_3	# of steps	V_1	V_2	V_3	# of steps	V_1	V_2	V_3
68	10	9	8	407	9	8	1	92	7	6	5
75	10	9	7	87	9	7	6	109	7	6	4
85	10	9	6	100	9	7	5	138	7	6	3
99	10	9	5	121	9	7	4	197	7	6	2
119	10	9	4	154	9	7	3	374	7	6	1
153	10	9	3	222	9	7	2	118	7	5	4
221	10	9	2	424	9	7	1	149	7	5	3
424	10	9	1	107	9	6	5	212	7	5	2
78	10	8	7	128	9	6	4	400	7	5	1
88	10	8	6	163	9	6	3	165	7	4	3
102	10	8	5	234	9	6	2	233	7	4	2
124	10	8	4	447	9	6	1	439	7	4	1
159	10	8	3	138	9	5	4	269	7	3	2
229	10	8	2	176	9	5	3	504	7	3	1
439	10	8	1	252	9	5	2	634	7	2	1

Continued

93	10	7	6	480	9	5	1	107	6	5	4
107	10	7	5	195	9	4	3	135	6	5	3
129	10	7	4	279	9	4	2	191	6	5	2
166	10	7	3	529	9	4	1	359	6	5	1
239	10	7	2	323	9	3	2	149	6	4	3
458	10	7	1	610	9	3	1	210	6	4	2
114	10	6	5	774	9	2	1	393	6	4	1
137	10	6	4	81	8	7	6	242	6	3	2
176	10	6	3	93	8	7	5	450	6	3	1
253	10	6	2	112	8	7	4	564	6	2	1
484	10	6	1	143	8	7	3	134	5	4	3
148	10	5	4	205	8	7	2	187	5	4	2
190	10	5	3	390	8	7	1	348	5	4	1
272	10	5	2	99	8	6	5	215	5	3	2
520	10	5	1	119	8	6	4	396	5	3	1
211	10	4	3	151	8	6	3	493	5	2	1
301	10	4	2	216	8	6	2	187	4	3	2
574	10	4	1	411	8	6	1	342	4	3	1
350	10	3	2	128	8	5	4	423	4	2	1
664	10	3	1	163	8	5	3	351	3	2	1
844	10	2	1	232	8	5	2				
74	9	8	7	440	8	5	1				
83	9	8	6	180	8	4	3				
96	9	8	5	256	8	4	2				
115	9	8	4	484	8	4	1				
148	9	8	3	296	8	3	2				
212	9	8	2	557	8	3	1				

For example, it will take minimum 68 steps with vulnerability configuration of $V_1 = 10$, $V_2 = 9$, $V_3 = 8$ for the attacker to reach the final goal with probability one. The largest number of steps for the attacker to achieve his goal is 844 steps by using the vulnerabilities, $V_1 = 10$, $V_2 = 2$ and $V_3 = 1$, with probability one.

4.3. Expected Path Length (EPL) Analysis

As described under Section 3.1.2 we measure the expected number of steps the attacker will take starting from the initial state to compromise the security goal. In **Table 3**, we present the calculations of the Expected Path Length of the attacker for various combinations of the vulnerabilities V_1, V_2 and V_3.

For example, it will take 8.25 EPL with vulnerability configuration of $V_1 = 10$, $V_2 = 9$, $V_3 = 8$ for the attacker to compromise the security goal. The largest Expected Path Length of the attacker is 72.8 using $V_1 = 8$, $V_2 = 2$ and $V_3 = 1$.

5. Development of the Statistical Models

The primary objective here is to utilize the information that we have calculated to develop a statistical model to predict the minimum number of steps to reach the stationary matrix and EPL of the attacker. We used the application software package "*R*" [15] for required calculations in developing these models.

5. 1. Developing a Statistical Model to Predict the Minimum Number of Steps

By using the information in **Table 2**, we developed a statistical model that estimates the minimum number of

Table 3. Expected path length for several vulnerabilities.

Expected path length	V_1	V_2	V_3	Expected path length	V_1	V_2	V_3
8.25	10	9	8	34.25	9	3	2
8.98	10	9	7	63.25	9	3	1
9.96	10	9	6	79.91	9	2	1
11.33	10	9	5	9.53	8	7	6
13.39	10	9	4	10.78	8	7	5
16.81	10	9	3	12.65	8	7	4
23.67	10	9	2	15.77	8	7	3
44.22	10	9	1	22.01	8	7	2
9.32	10	8	7	40.72	8	7	1
10.33	10	8	6	11.39	8	6	5
11.75	10	8	5	13.36	8	6	4
13.87	10	8	4	16.64	8	6	3
17.42	10	8	3	23.19	8	6	2
24.5	10	8	2	42.86	8	6	1
45.75	10	8	1	14.35	8	5	4
10.81	10	7	6	17.85	8	5	3
12.29	10	7	5	24.85	8	5	2
14.5	10	7	4	45.85	8	5	1
18.19	10	7	3	19.67	8	4	3
25.57	10	7	2	27.33	8	4	2
47.71	10	7	1	50.33	8	4	1
13	10	6	5	31.48	8	3	2
15.33	10	6	4	57.82	8	3	1
19.22	10	6	3	72.8	8	2	1
27	10	6	2	10.57	7	6	5
50.33	10	6	1	12.35	7	6	4
16.5	10	5	4	15.32	7	6	3
20.67	10	5	3	21.27	7	6	2
29	10	5	2	39.1	7	6	1
54	10	5	1	13.25	7	5	4
22.83	10	4	3	16.42	7	5	3
32	10	4	2	22.75	7	5	2
59.5	10	4	1	41.75	7	5	1
37	10	3	2	18.06	7	4	3
68.67	10	3	1	24.98	7	4	2
87	10	2	1	45.73	7	4	1
8.798	9	8	7	28.7	7	3	2
9.73	9	8	6	52.37	7	3	1
11.04	9	8	5	65.67	7	2	1
13	9	8	4	12.14	6	5	4
16.27	9	8	3	14.97	6	5	3
22.82	9	8	2	20.64	6	5	2
42.44	9	8	1	37.64	6	5	1

Continued

10.18	9	7	6	16.43	6	4	3
11.54	9	7	5	22.6	6	4	2
13.58	9	7	4	41.1	6	4	1
16.99	9	7	3	25.89	6	3	2
23.79	9	7	2	46.89	6	3	1
44.22	9	7	1	58.5	6	2	1
12.2	9	6	5	14.78	5	4	3
14.35	9	6	4	20.19	5	4	2
17.93	9	6	3	36.44	5	4	1
25.1	9	6	2	23.04	5	3	2
46.6	9	6	1	41.38	5	3	1
15.43	9	5	4	51.29	5	2	1
19.26	9	5	3	20.14	4	3	2
26.93	9	5	2	35.81	4	3	1
49.93	9	5	1	44	4	2	1
21.26	9	4	3	36.6	3	2	1
29.67	9	4	2				
54.92	9	4	1				

Table 4. Parametric Model: R^2 and adjusted R^2 values.

Model	R^2	Adjusted R^2
$Y_1 = 344.167 + 35.284V_1 - 34.115V_2 - 67.803V_3$	0.7244	0.7173
$Y_2 = 446.865 + 67.645V_1 - 81.662V_2 - 149.982V_3 - 1.24V_1V_2 - 13.7V_1V_3 + 29.354V_2V_3$	0.8835	0.8773
$Y_3 = 689.84 + 51.177V_1 - 138.815V_2 - 328.093V_3 - 0.3626V_1V_2 + 9.29V_1V_3 + 39.114V_2V_3 - 0.084V_1^2 + 8.479V_2^2 + 17.96V_3^2 - 3.47V_1V_2V_3$	0.9428	0.9376

steps the attacker takes to reach the goal state with probability one.

The quality of the model is measured by R^2 and adjusted R^2 values as defined below:

The first model in **Table 4** does not include interactions of the three Vulnerabilities, V_1, V_2 and V_3 and R^2 and R^2_{adj} reflect its quality of 0.7244 and 0.7173. The second model shows that there is a significant binary interaction of the each factors and the statistical model shows a significant improvement with R^2 and R^2_{adj} of 0.8835 and 0.8773 respectively. However, the best statistical model is obtained when we consider in addition to individual contributions of V_1, V_2 and V_3, two way and three way significant interactions. Thus, from the above table the third model with R^2 and R^2_{adj} of 0.9428 and 0.9376 respectively attest to the fact that this statistical model is excellent in estimating the minimum number of steps that an attacker will need to achieve his goal.

5.2. Developing a Parametric Model to Predict the Expected Path Length

By using **Table 3** results we developed a model to find the Expected Path Length that the attacker will take starting from the initial state to reach the security goal.

To utilize the quality of the model we use R^2 concept and by comparing the values in **Table 5**, the third model gives the highest R^2 and adjusted R^2 value. Therefore we can conclude that the third model gives the best prediction of EPL.

5.3. Comparison of Parametric/Statistical Model Value with Markov Model Value

From the comparison shown in **Table 6**, we can conclude that our proposed statistical model gives accurate predictions.

5.4. Rank of Attributable Variables

In **Table 7**, below we present the ranks of the most important attributable variables with respect to their contribution to estimate the EPL.

The most attributable variable (vulnerability) is V_3 in quadratic form and individually. Whereas the minimum risk factor is the vulnerability V_1. Thus, one can use this ranking to take precautionary measures addressing the most dangerous vulnerability or vulnerabilities with priority.

Table 5. Parametric model (EPL): R^2 and adjusted R^2 values.

Model	R^2	Adjusted R^2
$Y_1 = 35.975 + 3.622V_1 - 3.497V_2 - 6.845V_3$	0.7253	0.7181
$Y_2 = 46.301 + 6.904V_1 - 8.28V_2 - 15.178V_3 - 0.128V_1V_2 - 1.384V_1V_3 + 2.97V_2V_3$	0.8839	0.8778
$Y_3 = 70.62 + 5.338V_1 - 14.108V_2 - 33.144V_3 - 0.041V_1V_2 + 0.942V_1V_3 + 3.943V_2V_3 - 0.015V_1^2 + 0.864V_2^2 + 1.814V_3^2 - 0.35V_1V_2V_3$	0.943	0.9378

Table 6. Error calculation of parametric/statistical model (EPL) and Markov model.

Parametric value	Markov Value	Error
9.099	9.96	0.861
43.596	44.22	0.624
61.487	63.25	1.763
39.62	42.86	3.24
49.91	51.29	1.38
43.68	44	0.32
10.49	10.57	0.08

Table 7. Ranking the variables according to contribution.

Variable	Rank
V_3^2	1
V_3	2
V_2	3
V_2^2	4
V_2V_3	5
$V_1V_2V_3$	6
V_1	7
V_1V_3	8
V_1V_2	9
V_1^2	10

6. Conclusions

We have developed a very accurate statistical model that can be utilized to predict the minimum steps to reach the goal state and predict the expected path length.

This developed model can be used to identify the interaction among the vulnerabilities and individual variables that drive the EPL.

We ranked the attributable variables and their contribution in estimating the subject length. By using these rankings, security administrators can have a better knowledge about priorities. This will help them to take the necessary actions regarding their security system.

Here we develop a model for three vulnerabilities and we can expand this model to any large

Network System. Thus, the proposed methods will assist in making appropriate security decisions in advance.

References

[1] Secunia Vulnerability Review 2015: Key Figures and Facts from a Global Information Security Perspective. https://secunia.com/?action=fetch&filename=secunia_vulnerability_review_2015_pdf.pdf

[2] NVD, National Vulnerability Database. http://nvd.nist.gov/

[3] Kijsanayothin, P. (2010) Network Security Modeling with Intelligent and Complexity Analysis. PhD Dissertation, Texas Tech University.

[4] Alhazmi, O.H., Malaiya, Y.K. and Ray, I. (2007) Measuring, Analyzing and Predicting Security Vulnerabilities in Software Systems. *Computers and Security Journal*, **26**, 219-228.

[5] Noel, S., Jacobs, M., Kalapa, P. and Jajodia, S. (2005) Multiple Coordinated Viewsfor Network Attack Graphs. *VIZSEC*'05: *Proc. of the IEEE Workshops on Visualization for Computer Security*, Minneapolis, October 2005, 99-106.

[6] Mehta, V., Bartzis, C., Zhu, H., Clarke, E.M. and Wing, J.M. (2006) Ranking Attack Graphs. In: Zamboni, D. and Krugel, C., Eds., *Recent Advances in Intrusion Detection*, Vol. 4219, 127-144. http://dx.doi.org/10.1007/11856214_7

[7] Frei, S. (2009) Security Econometrics: The Dynamics of (IN) Security. PhD Dissertation, ETH, Zurich.

[8] Schiffman, M. Common Vulnerability Scoring System (CVSS). http://www.first.org/cvss/

[9] Bass, T. (2000) Intrusion Detection System and Multi-Sensor Data Fusion. *Communications of the ACM*, **43**, 99-105.

[10] Lawler, G.F. (2006) Introduction to Stochastic Processes. 2nd Edition, Chapman and Hall/CRC Taylor and Francis Group, London, New York.

[11] Jajodia, S. and Noel, S. (2005) Advanced Cyber Attack Modeling, Analysis, and Visualization. 14*th USENIX Security Symposium*, Technical Report 2010, George Mason University, Fairfax.

[12] Abraham, S. and Nair, S. (2014) Cyber Security Analytics: A Stochastic Model for Security Quantification Using Absorbing Markov Chains. *Journal of Communications*, **9**, 899-907.

[13] Wang, L., Singhal, A. and Jajodia, S. (2007) Measuring Overall Security of Network Configurations Using Attack Graphs. *Data and Applications Security XXI*, **4602**, 98-112.

[14] Wang, L., Islam, T., Long, T., Singhal, A. and Jajodia, S. (2008) An Attack Graph-Based Probabilistic Security Metric. DAS 2008, LNCS 5094, 283-296.

[15] R statistics Tool. http://www.r-project.org

Appendix

Model 1R results

1) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$

Here y—# of steps takes to reach the goal state with highest probability
X_i—Vulnerabilities
b_i—coefficient

Coefficients:

	Estimate	Std. Error	t value	Pr (>|t|)
(Intercept)	344.167	41.154	8.363	1.55e−13***
X1	35.284	5.984	5.896	3.74e−08***
X2	−34.115	5.984	−5.701	9.21e−08***
X3	−67.803	5.984	−11.331	<2e−16***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 90.95 on 116 degrees of freedom
Multiple R-squared: 0.7244, Adjusted R-squared: 0.7173
F-statistic: 101.6 on 3 and 116 DF, p-value: <2.2e−16

2) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_1X_2 + b_5X_1X_3 + b_6X_2X_3$

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	446.865	72.410	6.171	1.09e−08***
X1	67.645	9.772	6.922	2.85e−10***
X2	−81.662	23.169	−3.525	0.000613***
X3	−149.982	29.943	−5.009	2.04e−06***
X4	−1.240	2.516	−0.493	0.623005
X5	−13.700	3.863	−3.546	0.000570***
X6	29.354	2.516	11.667	<2e−16***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 59.9 on 113 degrees of freedom
Multiple R-squared: 0.8835, Adjusted R-squared: 0.8773
F-statistic: 142.9 on 6 and 113 DF, p-value: <2.2e−16

AIC = 1371.591

3) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_1X_2 + b_5X_1X_3 + b_6X_2X_3 + b_7X_1^2 + b_8X_2^2 + b_9X_3^2 + b_{10}X_1X_2X_3$

Call:
lm(formula = y ~ X)
Residuals:

Min	1Q	Median	3Q	Max
−119.916	−25.326	5.661	26.622	110.223

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	689.84236	105.66582	6.529	2.17e−09***
X1	51.17739	23.71769	2.158	0.033141*
X2	−138.81536	26.81969	−5.176	1.04e−06***

X3	−328.09288	58.78621	−5.581	1.76e−07***
X4	−0.36269	3.50341	−0.104	0.917737
X5	9.29187	6.64327	1.399	0.164745
X6	39.11435	10.51884	3.719	0.000318***
X7	−0.08396	1.86012	−0.045	0.964079
X8	8.47917	1.86012	4.558	1.35e−05***
X9	17.96149	1.86012	9.656	2.61e−16***
X10	−3.47455	1.07421	−3.235	0.001613**

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 42.74 on 109 degrees of freedom
Multiple R-squared: 0.9428, Adjusted R-squared: 0.9376
F-statistic: 179.7 on 10 and 109 DF, p-value: <2.2e−16

AIC = 1362.254

4) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_2X_3 + b_5X_2^2 + b_6X_3^2 + b_7X_1X_2X_3$

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	621.3262	32.8175	18.933	<2e−16***
X1	56.8012	4.0550	14.008	<2e−16***
X2	−132.3630	13.7273	−9.642	<2e−16***
X3	−253.7456	14.8482	−17.089	<2e−16***
X4	30.1674	4.6875	6.436	3.15e−09***
X5	7.3590	1.4064	5.233	7.88e−07***
X6	17.9615	1.8543	9.687	<2e−16***
X7	−2.3535	0.3204	−7.344	3.56e−11***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 42.61 on 112 degrees of freedom
Multiple R-squared: 0.9416, Adjusted R-squared: 0.9379
F-statistic: 258 on 7 and 112 DF, p-value: <2.2e−16

AIC = 1304.753

Model 2R results

1) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3$

Here y—# of steps takes to reach the goal state with highest probability
X_i—Vulnerabilities
b_i—coefficient

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	35.9750	4.1628	8.642	3.53e−14***
X1	3.6224	0.6053	5.985	2.47e−08***
X2	−3.4970	0.6053	−5.778	6.48e−08***
X3	−6.8457	0.6053	−11.310	<2e−16***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 9.199 on 116 degrees of freedom
Multiple R-squared: 0.7253, Adjusted R-squared: 0.7181
F-statistic: 102.1 on 3 and 116 DF, p-value: <2.2e−16

2) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_1X_2 + b_5X_1X_3 + b_6X_2X_3$

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	46.3018	7.3227	6.323	5.28e−09***
X1	6.9038	0.9882	6.986	2.08e−10***
X2	−8.2824	2.3430	−3.535	0.000592***
X3	−15.1780	3.0281	−5.012	2.01e−06***
X4	−0.1283	0.2544	−0.504	0.615103
X5	−1.3842	0.3907	−3.543	0.000577***
X6	2.9700	0.2544	11.673	<2e−16***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 6.058 on 113 degrees of freedom
Multiple R-squared: 0.8839, Adjusted R-squared: 0.8778
F-statistic: 143.4 on 6 and 113 DF, p-value: <2.2e−16

AIC = 821.6622

3) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_1X_2 + b_5X_1X_3 + b_6X_2X_3 + b_7X_1^2 + b_8X_2^2 + b_9X_3^2 + b_{10}X_1X_2X_3$

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	70.62069	10.68479	6.609	1.47e−09***
X1	5.33882	2.39830	2.226	0.028066*
X2	−14.10835	2.71197	−5.202	9.32e−07***
X3	−33.14449	5.94439	−5.576	1.81e−07***
X4	−0.04135	0.35426	−0.117	0.907303
X5	0.94165	0.67176	1.402	0.163826
X6	3.94315	1.06365	3.707	0.000331***
X7	−0.01535	0.18809	−0.082	0.935119
X8	0.86413	0.18809	4.594	1.17e−05***
X9	1.81443	0.18809	9.646	2.74e−16***
X10	−0.35045	0.10862	−3.226	0.001656**

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 4.322 on 109 degrees of freedom
Multiple R-squared: 0.943, Adjusted R-squared: 0.9378
F-statistic: 180.4 on 10 and 109 DF, p-value: <2.2e−16

AIC = 812.3033

4) $Y = b_0 + b_1X_1 + b_2X_2 + b_3X_3 + b_4X_2X_3 + b_5X_2^2 + b_6X_3^2 + b_7X_1X_2X_3$

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	64.04972	3.31940	19.296	<2e−16***

X1	5.80147	0.41015	14.145	<2e−16***
X2	−13.46770	1.38848	−9.700	<2e−16***
X3	−25.64100	1.50186	−17.073	<2e−16***
X4	3.05457	0.47413	6.442	3.05e−09***
X5	0.74725	0.14225	5.253	7.21e−07***
X6	1.81443	0.18755	9.674	<2e−16***
X7	−0.23834	0.03241	−7.353	3.41e−11***

Signif. codes: 0 "***" 0.001 "**" 0.01 "*" 0.05 "." 0.1 " " 1

Residual standard error: 4.31 on 112 degrees of freedom
Multiple R-squared: 0.9418, Adjusted R-squared: 0.9381
F-statistic: 258.8 on 7 and 112 DF, p-value: <2.2e−16

AIC = 754.87