Cascade Damage Estimation Model for Internet Attacks

Taek Lee[1], Hoh Peter In[1,*], Eul-Gyu Im[2], and Heejo Lee[1]

[1] Department of Computer Science and Engineering,
Korea University, Seoul, 136-713, Republic of Korea
{comtaek, hoh_in, heejo}@korea.ac.kr
[2] College of Information and Communications,
Hanyang University, Seoul, 133-791, Republic of Korea
imeg@hanyang.ac.kr

1 Introduction

Risk analysis and damage estimation are inevitable studies to gain essential data for making a better decision in security investment. The most reasonable metrics to measure the damage of a security accident are *recovery cost* and *business opportunity cost*[1,2,3,4]. In the case of a worm accident, the costs mean just the direct damage caused by infected systems. However, collaterally cascading damage is also serious damage which can impact on other innocent systems having depended on the infected systems for the purpose of processing their business or demanding some service.

2 Our Proposed Damage Estimation Model

Cascade Damage Estimation Model(CDEM) is represented by a graph-based, so called Dependency Tree Diagram(DTD), algorithm to be able to identify dependence relation between systems and calculate the potential cascading damage caused by the business-dependent relation with an infected system. Given a domain, the algorithm(table 2) quantitatively estimates the degree of loss and its likelihood by a graphic and probabilistic approach.

Table 1. The definition of DTD

DTD=<*N*,*E*> *N* is a set of nodes, $N=\{n_1,n_2,n_3,\dots\}$ *E* is a set of edges, $E=\{e_{i,j} \mid i\neq j \wedge n_i,n_j \in N\}$ $n_i=<FP,OC,CD>$, $e_{i,j}=<BD,CP>$	*FP* : business failure probability *OC* : business opportunity cost *CD* : calculated cascade damage *BD* : business performance degrading rate *CP* : dependency connection probability

* Corresponding author.

H. Chen et al. (Eds.): WISI 2006, LNCS 3917, pp. 163 – 164, 2006.

Table 2. Cascade Damage Estimation Algorithm

```
Cascading Damage Estimation Algorithm ()
{
 Total_CD = 0
 make_DTD_structure()
 D = {"nodes damaged directly by a worm"}
 X = N – D
 for each x in X
 {
  x.FP = 1
  Y = {"all parent nodes of node x"}
  for each y in Y
  {
   if (y.FP is not defined) { y.FP ← find_cp(y) }
   Likelihood ← y.FP × e_y,x.CP
   if (AND-case) x.FP ← x.FP × Likelihood
   elseif (OR-case) x.FP ← x.FP × (1-Likelihood )
    Loss ← e_y,x.BD × x.OC
    x.CD ← x.CD + Loss × Likelihood
  }
  if (OR-case) x.FP = 1 - x.FP
  Total_CD ← Total_CD + x.CD
 }
}
```

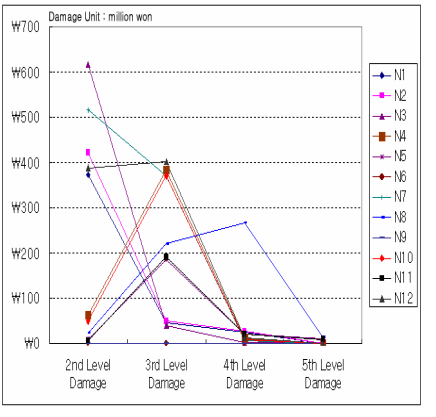

Fig. 1. Cascade damage in each level

3 Evaluation and Application of CDEM

The proposed algorithm was tested by random experiment simulation approach in order to check its validation in terms of calculating damage probability in each node. As the result, we could guarantee the consistency of our estimation algorithm. CDEM can be utilized not only in estimating cascade damage(figure 1) but also in identifying critical systems and hence analyzing Return On Security Investment(ROSI). The prioritization information of the cascade damages triggered by each causal node can be a good evidence to look for weak points on the infrastructure consisting of system nodes, in other words, the best promising points to be defended and invested in the perspective of cascade damage prevention.

References

[1] Incident Cost Analysis and Modeling Report I, II, Committee on Institutional Cooperation, 2000
[2] "Information Security Incident Survey and Damage Calculation Model", Japan Network Security Association, March 31, 2004
[3] Nicholas Weaver, Vern Paxson, "A Worst-Case Worm", May 5, 2004
[4] Thomas Dubendorfer, Arno Wagner, Bernhard Plattner, "An Economic Damage Model for Large-Scale Internet Attacks", WET ICE'04