A model of Internet and a method to suppress virus

Li Chen
Fudan University

Abstract

Internet is a large network with a special topology. Better modeling its structure is the key to finding virus spread pattern and suppressing virus spread efficiently. In my research, I find the Internet features a division of transit domain layer and stub domain layer, and propose a model of two layers using different matrix to describe their structure, vulnerability and value. According to the indexes in my model, I find an overall best solution to prevent virus spread and ensure network connectivity. In my simulation based on a 100 nodes Internet pattern network, it receives cheerful results.
Keywords: network, Internet, virus spread, virus suppression, model

Introduction

Landmark network models

After the establishment of two landmark network models—the small-world models[1] and the scale-free networks[2], scientists have broken through the restriction of random graphs theory[3] which established by Hungary mathematician P. Erdos and Re´nyi nearly half century ago, and have gained great achievement.

Previous research has found that many networks in our real world follow the small-world and scale-free model such as nerve system in our brain, the power grid network, the airlines network, cooperation in research and all kinds of political, economic, social connection networks etc.[4]

The network structure can be characterized by the connectivity properties between nodes. The degree of a node is the number of its links connected to other nodes. According to the degree distribution, which is the probability a certain degree node takes percentage in all the nodes in a network, complex networks can be classified into homogeneous and heterogeneous (or Inhomogeneous) networks. Homogeneous networks such as random graphs and small-world models possess binomial or Poisson degree distribution, where the degrees concentrate around the mean degree. Heterogeneous networks such as scale-free networks have a heavy-tailed degree distribution that approximately follows a power law [5].

Scale-free networks, with power-law degree distribution, are remarkably resistant to random failure [6,7], for most links in the network are connected to a few large-degree nodes, and the failure of these peripheral nodes causes much slighter danger to the whole network. However, when there's an intentional attack, the failure of a few hubs (large degree nodes) may trigger catastrophic damage to the whole network. So the scale-free model is both robust and fragile.

Internet topology

From Professor Li's course, we know the Internet also follows scale-free model. The node distribution probability has an exponential degree distribution -- $P(k) \propto k^{-2.2}$ (k is the degree of nodes, P (k) is the node distribution probability). With more and more people stepping into the Internet era, nodes in that massive network increase dramatically as years roll on. As we can see in Scale-free model, the direct result is some super nodes have larger and larger degree, which leaves us with a serious problem that if these nodes have been attacked by some rapidly spreading virus, what will happen to our Internet?

First of all, let us have a view of the Internet structure. The Internet is an ever growing large network. Such a tremendously large and complicated network has some basic characters should be paid attention to when we research its robustness.

Internet can be viewed as a collection of interconnected routing domains. Each routing domain is a group of nodes (routers, switches, and hosts), under a single administration that share routing information and policy. Each routing domain in the Internet can be classified as either a stub or transit domain. A stub domain carries only traffic that originates or terminates in the domain. Transit domains is to interconnect stub domains efficiently. Stub domains generally correspond to campus networks or other collections of interconnected local area networks (LANs), while transit domains are almost always wide or metropolitan area networks (WANs or MANS). A transit domain consists of a set of backbone nodes connected to a number of stub domains, via gateway nodes in the stubs. (Fig. 1) [8]

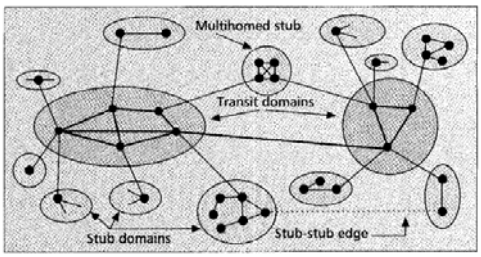


Figure 1. *An example of the Internet domain structure.*[1]

Revelation from the research paper can be proved in our real world. Take our dormitory network as an example. In our dormitory, every electronic device (mobile phone, laptop, iPad, etc.) is connected to a router in every room, which is connected to the switch in every flour. Then the switch is connected to a larger switch in every building, which finally connects to the host in dorm living area, through China Telecom connecting to Internet. While, few direct connections are established between different groups of that hierarchy structures, such as connections between Fudan and Shanghai Jiao tong University.

From the description above, we can conclude these characters.

First, lower level of the Internet topology is highly like a hierarchy structure. And the connections inside these stub domains are much denser than outside. So imagining a virus spreading in a stub domain, all the nodes inside are highly susceptible.

Second, the gateway nodes play an essential role of gates connecting the whole stub domain to the Internet and passing data to its child nodes. When gateway nodes are infected, the whole stub domain is in great danger.

[1] Figure is from Kenneth L. Calvert, Matthew B. Doar, Ellen W. Zegura. Modeling Internet Topology. Global Telecommunications Conference, 1996. GLOBECOM '96. 'Communications: The Key to Global Prosperity, 161.

Third, a few edges which connected directly between nodes from different stub domains should not be neglected. They may exist in our Internet as email, P2P connections, bringing us convenience but leave us with risks of being infected.

In conclusion, our Internet is more similar to Local-world model [9], which is an advanced version of BA model where nodes connect preferentially but are more likely to be connected in a separate group, while connections between groups are sparse.

Tools to use

Gephi 0.8.2, a network simulator, visualizer and analyzer.

createpajek , a net format file creator in pajeck200.

GDTANG (The Geographic Directed Tel Aviv University Network Generator) [10], a Perl program that generates synthetic Internet-like topologies using an improved BA type model. It produces networks with a power-law degree distribution, realistic maximal degrees and "Dense Core", accurate number of leaves and geographically meaningful clusters.

Microsoft Office Excel 2013 to do data analysis and draw basic graph.

Matlab R2012a to do matrix calculation.

Models

Model of the Internet

With the help of GDTANG, I created an Internet model with 100 nodes and no double edges. Other network generating parameters use default values. Importing data to Gephi, after some location adjustment, the network displays in Fig. 2 and 3.

The GDTANG model graph corroborates the Internet structure again. We can clearly find the branch shape hierarchy structure under these relatively large nodes on Fig. 2, some typical ones are branches under node 1, 8 and 12 in Fig. 3. Nodes representing transit domains are in the middle, which are highly interconnected with each other. We can also find that almost all stubs are multi-homed stubs and stub-stub connections are not rare such as connection between node 53 and 96 in Fig. 3.

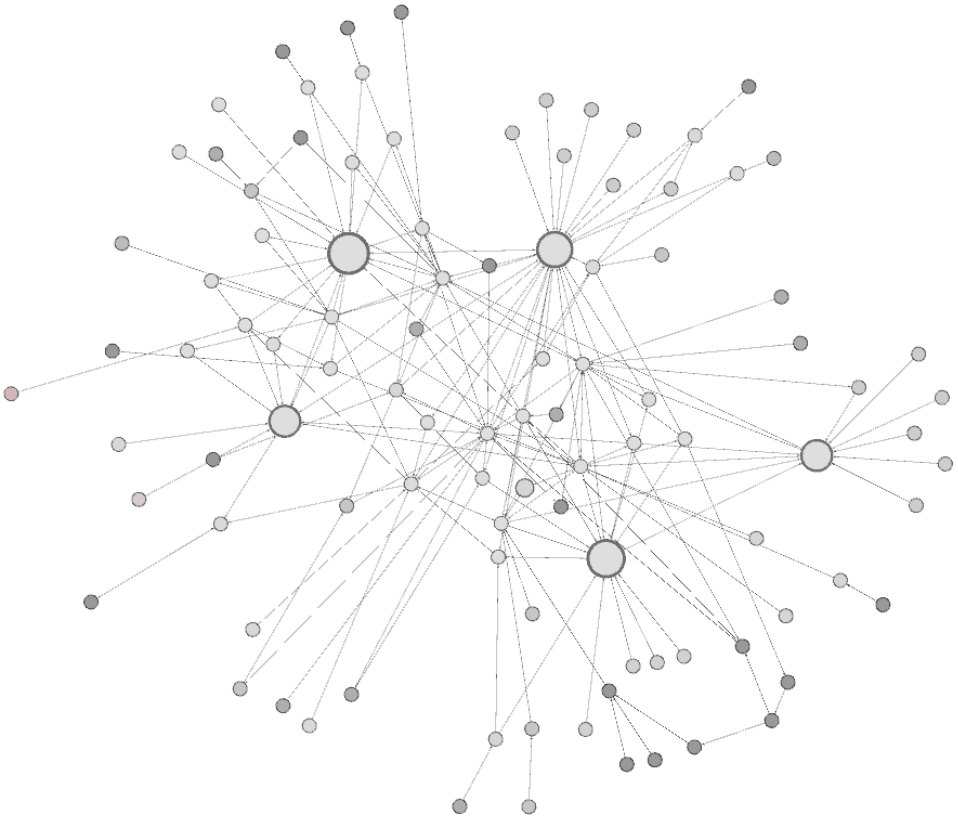


Figure 2. *An Internet model with 100 nodes and 211 edges with no double edges generated by GDTANG.*

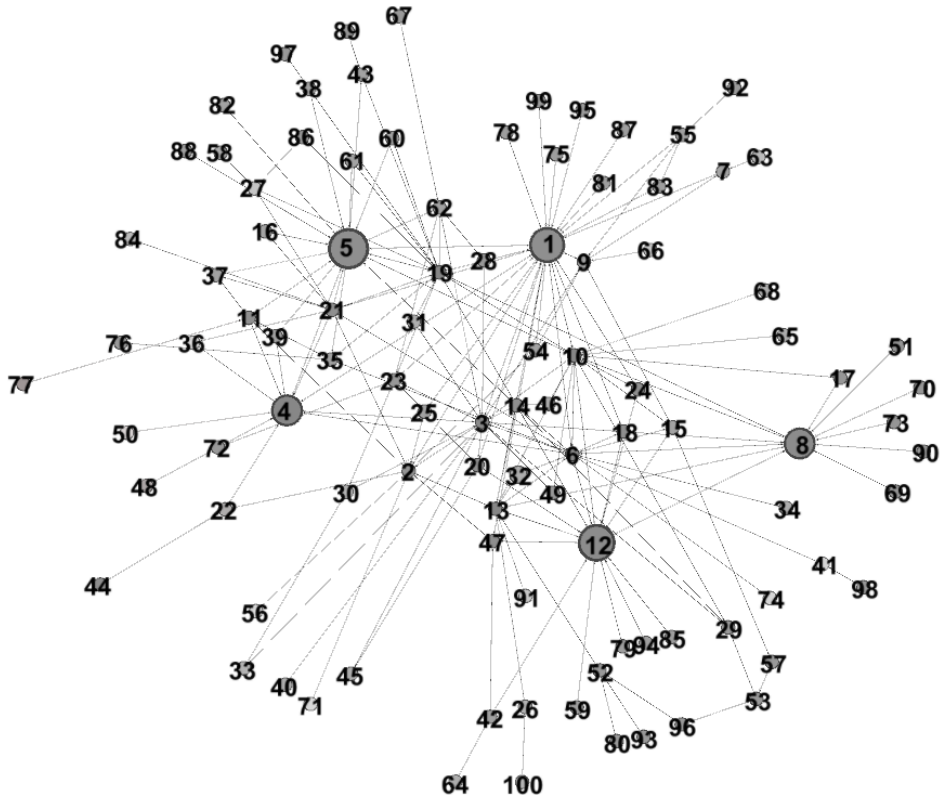


Figure 3. *An Internet model with 100 nodes and 211 edges with no double edges generated by GDTANG. (With node labels)*

Having revealed the basic structure and character in Internet, we can assure the GDTANG model is valid and can be used in further research.

According to the Internet topology, I think we can divide the Internet into two layers. The nodes within a stub domains are the lower layer, while connections within transit domains are the higher one. Their different topologies determine that varied models and diverse methods should be taken to analyze and suppress the virus from spreading.

Model of valuing hierarchy structure

The structure of the Internet determines that virus spread is in group. That is when the gateway nodes are attacked, computers in the same LAN or under the same router are highly vulnerable. So the gateway nodes not only control information exchange, but also hold the safety of the whole group.

But, with so many stub domains in Internet, we need to find a method to assess the value of a stub domain with its scale, vulnerability and importance in consideration so that we can preferentially protect or immune or cure these valuable stub domains.

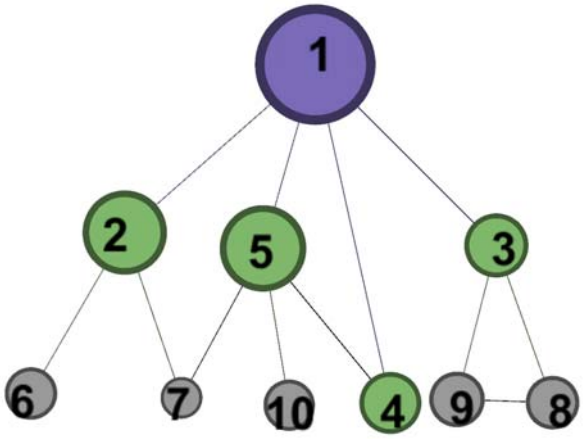


Figure 4. *An example of hierarchy structure for matrix calculation. The large blue node is the gateway node, the green and grey ones are its child nodes. Numbers in the circle is its ID.*

In my research, I define a matrix to describe links in a hierarchy structure.

*Definition 1: A is an m*m matrix describes the connectivity of a network, m is the total number in a network, a_{ij} =1 means node i and node j have been connected.*

$$A = \begin{bmatrix} a_{11}, a_{12}, \dots\dots, a_{1m} \\ a_{21}, a_{22}, \dots\dots, a_{2m} \\ \dots\dots\dots\dots\dots\dots\dots \\ a_{m1}, a_{m2}, \dots\dots, a_{mm} \end{bmatrix}$$

Possibility to be infected is not the same to each nodes. To have a clear view of that possibility, I define another matrix.

*Definition 2: P is an m*m matrix describes the infection possibility between two nodes in a network, m is the total number in a network, p_{ij} means node i has a percentage of p_{ij} spreading virus to node j.*

$$P = \begin{bmatrix} p_{11}, p_{12}, \dots\dots, p_{1m} \\ p_{21}, p_{22}, \dots\dots, p_{2m} \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ p_{m1}, p_{m2}, \dots\dots, p_{mm} \end{bmatrix}$$

Each nodes have different values. For example, bank account computer is much more important than an entertainment computer, so protection level should be different. I also use a matrix to mark their difference.

*Definition 3: V is an m*1 matrix describes the value (or importance) of each nodes, m is the total number in a network, v_i means the damage of node i being infected.*

$$\mathrm{V} = \begin{bmatrix} v_1 \\ v_2 \\ \dots \\ \dots \\ v_m \end{bmatrix}$$

With these matrixes, we can have a rough view of how much damage an infected node can cause to other nodes.

*Definition 4: C is an m*1 matrix describes the total value of each node when all its neighbors are infected, m is the total number in a network, c_i means the total cost of damage it causes when all node i's neighbors are infected.*

$$\mathrm{C} = \mathrm{A}.\cdot \mathrm{P} \cdot \mathrm{V} = \begin{bmatrix} a_{11} \cdot p_{11}, a_{12} \cdot p_{12}, \dots\dots, a_{1m} \cdot p_{1m} \\ a_{21} \cdot p_{21}, a_{22} \cdot p_{22}, \dots\dots, a_{2m} \cdot p_{2m} \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ a_{m1} \cdot p_{m1}, a_{m2} \cdot p_{m2}, \dots\dots, a_{mm} \cdot p_{mm} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ \dots \\ \dots \\ v_m \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ \dots \\ \dots \\ c_m \end{bmatrix}$$

$$c_i = a_{i1} * v_1 + a_{i2} * v_2 + \cdots + a_{im} * v_m$$

Among c_i, one of them is of more revelation value.

Definition 5: When i=1 is the gateway node, call GSD = c_1 the gateway spread damage. It reveals the initial damage caused by gateway nodes being infected.

When the number of total node levels is low, GSD usually has a large number.

Definition 6: The sum of c_i in a matrix C is the total damage the whole network is infected, calling it Complete spread damage (CSD)

$$\mathrm{CSD} = \sum_{i=1}^{m} ci$$

Definition 7: The percentage of GSD among CSD is called gateway penetration ratio (GPR), which reflects how much damage virus has caused when the top level of a hierarchy structure is infected.

$$\mathrm{GPR} = \frac{GSD}{CSD} \times 100\%$$

GPR gives us suggestion whether that stub is still worth rescuing. If it is high enough, abandon that stub domain and rescue or protect other stub domains may be a better choice.

But we still do not consider the node quantities and connection patterns in the stub domain. To have a view of that, we need to find another index.

Definition 8: Starting from gateway node, which marked as level 1. Find all its child nodes, which marked as level 2. Find all level 2 nodes' child nodes, mark them as level 3. Do that kind of search till the bottom level node which marked as level n. If some nodes have already been marked a number, ignore them.

*D is an m*m matrix to describe connection probability between different levels of nodes in a hierarchy structure,*

$$D = A.\cdot P.\cdot V$$

*Sum up all the links with start nodes in the same level to end nodes in the same level, arrange them in an n*n matrix Q.*

$$Q = \begin{bmatrix} q_{11}, q_{12}, \dots \dots, q_{1n} \\ q_{21}, q_{22}, \dots \dots, q_{2n} \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ q_{n1}, q_{n2}, \dots \dots, q_{nn} \end{bmatrix}$$

q_{ij} means total virus spread possibility that level i nodes spread to level j nodes.

Q describes the virus spread capability from a level to another.

When we review a hierarchy structure, we care different levels in a different perspective. For example, in this virus spread issue, higher level connections should be paid more attention than lower levels. A manually changeable matrix should be defined to represent our perspective.

*Definition 9: Matrix F (n*n) is a set of parameters we choose to reflect weights between different levels of a hierarchy network.*

$$\mathrm{F} = \begin{bmatrix} X & H & X \\ H & M & L \\ X & L & L \end{bmatrix}$$

(Taking level n = 3 as an example)

X means no possibility to spread (from level 1 to level 3), H means we pay high attention to that kind of spread (from level 1 to level 2), so we can choose H = 0.7, for example; M means we pay medium attention to that kind of spread (from level 2 to level 2), so we can choose M=0.2, for example; L means it is a trivial connection between such levels (from level 2 to level 3), choose L=0.1, for example.

With F defined, we can have a relatively accurate and suitable index to view a hierarchy network.

Definition 10: Hierarchy network infection index (HNII) is an index to describe a hierarchy network considering its connection between levels, number of connections, degrees of nodes, virus spread success rate, nodes importance in a specific way determined by F.

$$\mathrm{HNII} = \sum_{a_{ii}} Q \cdot F$$

$\sum_{a_{ii}} X$ means adding all diagonal numbers in matrix X..

Example of definition 1 to 10 using Fig. 4

$$\mathbf{A} = \begin{bmatrix} 0&1&1&1&1&0&0&0&0&0 \\ 1&0&0&0&0&1&1&0&0&0 \\ 1&0&0&0&0&0&0&1&1&0 \\ 1&0&0&0&1&0&0&0&0&0 \\ 1&0&0&1&0&0&1&0&0&1 \\ 0&1&0&0&0&0&0&0&0&0 \\ 0&1&0&0&1&0&0&0&0&0 \\ 0&0&1&0&0&0&0&0&1&0 \\ 0&0&1&0&0&0&0&1&0&0 \\ 0&0&0&0&1&0&0&0&0&0 \end{bmatrix}$$

P uses matlab function P=rand(10).

0.75774	0.823458	0.489764	0.498364	0.959291	0.349984	0.285839	0.129906	0.601982	0.825817
0.743132	0.694829	0.445586	0.959744	0.547216	0.196595	0.7572	0.568824	0.262971	0.538342
0.392227	0.317099	0.646313	0.340386	0.138624	0.251084	0.753729	0.469391	0.654079	0.996135
0.655478	0.950222	0.709365	0.585268	0.149294	0.616045	0.380446	0.011902	0.689215	0.078176
0.171187	0.034446	0.754687	0.223812	0.257508	0.473289	0.567822	0.337123	0.748152	0.442678
0.706046	0.438744	0.276025	0.751267	0.840717	0.35166	0.075854	0.162182	0.450542	0.106653
0.031833	0.381558	0.679703	0.255095	0.254282	0.830829	0.05395	0.794285	0.083821	0.961898
0.276923	0.765517	0.655098	0.505957	0.814285	0.585264	0.530798	0.311215	0.228977	0.004634
0.046171	0.7952	0.162612	0.699077	0.243525	0.549724	0.779167	0.528533	0.913337	0.77491
0.097132	0.186873	0.118998	0.890903	0.929264	0.917194	0.934011	0.165649	0.152378	0.817303

V use matlab function x=rand(10),V=x(:,1).

V=[0.9797;0.4389;0.1111;0.2581;0.4087;0.5949;0.2622;0.6028;0.7112;0.2217]

C=[0.9365; 1.0436;1.1324; 0.7032; 0.4725;0.1926;0.2714;0.2356;0.3367;0.3798]

GSD = 0. 9365

CSD = 5.7043

GPR = 16.42%

$$Q = \begin{bmatrix} 0 & 2.71 & 0 \\ 0.61 & 0.13 & 0.95 \\ 0 & 1.14 & 0.51 \end{bmatrix}$$

$$F = \begin{bmatrix} 0 & 0.7 & 0 \\ 0.7 & 0.2 & 0.1 \\ 0 & 0.1 & 0.1 \end{bmatrix}$$

$$HNII = 2.611$$

Matlab code and raw data can be found in attached files.

With GPR and HNII, we model the whole hierarchy network into these two index values. GPR reveals its gateway influence factor, HNII reveals its level structure, dimension, value and vulnerability. The high GPR means its initial spread is dramatic, so there's no need rescuing other nodes not yet been infected. The high HNII means that hierarchy structure is low in number of levels, large in scale, high in node value, so easy to be spread fast and need special care and immunization as soon as possible.

Model of transit domain structure

After the hierarchy model, we largely reduce the size of the internet model, turning every branch of nodes into one gateway node with GPR and HNII. Links in transit level are dense and complicated, the best way to describe that network is to use matrix to describe its connectivity.

Similar to *Definition 1 and 2*, we can also calculate its $\mathrm{A}\cdot \mathrm{P}$ according to its connectivity and possibility to spread.

*Definition 11: E is a k*k matrix to describe every spread possibility between two nodes in a network. k is the total number of nodes in the network.*

e_i is the total sum of spread possibility to node i.

$$\mathrm{E} = \mathrm{A}\cdot \mathrm{P} = \begin{bmatrix} a_{11}\cdot p_{11}, a_{12}\cdot p_{12}, \dots\dots, a_{1k}\cdot p_{1k} \\ a_{21}\cdot p_{21}, a_{22}\cdot p_{22}, \dots\dots, a_{2k}\cdot p_{2k} \\ \dots\dots\dots\dots\dots\dots\dots \\ a_{k1}\cdot p_{k1}, a_{k2}\cdot p_{k2}, \dots\dots, a_{kk}\cdot p_{kk} \end{bmatrix}$$

$$e_i = a_{1i}\cdot p_{1i} + a_{2i}\cdot p_{2i} + \cdots + a_{ki}\cdot p_{ki}$$

So we get another important index to consider these gateway nodes. As we can see from the *definition 11,* e_i is highly related to the structure of the transit domain network. The more effective spread path a node has, the higher value it is. Cutting some edges between nodes can sharply decrease its value.

Model of spreading

Two generally used models to describe epidemics are SIS and SIR models.

S represents susceptible group, I represents infected ones, R represents recovered or removed ones.

In SIS model, individuals are considered to recover with no immunity to the disease.

$$\frac{dS}{dt} = -\alpha IS + \gamma I$$

$$\frac{dI}{dt} = \alpha IS - \gamma I$$

In SIR model, individuals once recovered will immune to the disease.

$$\frac{dS}{dt} = -\alpha IS$$

$$\frac{dI}{dt} = \alpha IS - \gamma I$$

$$\frac{dR}{dt} = \gamma I$$

Rate of α is considered the contact or infection rate of the disease, γ is considered the recovery rate of disease [11].

In the situation of computer virus, the SIR model is more appropriate to the real fact. For whenever a computer install a certain patch or take actions toward a loophole, the same virus is unable to attack again.

But some aspects SIR model does not suit well in virus spread situation is that the α and γ may not the same to each connection. For there is a variety of network topology and different attack methods. The firewall and anti-virus system are not the same to each computer as well. So in my model, I introduce the infection possibility matrix P to mark the difference.

*Definition 12: M is the 1*k matrix describe the recovery or immune of nodes. m_i=0 means it is recovered or immunized.*

$$M = [m_1, m_2, \dots \dots m_k\ ]$$

*Definition 13: The infected nodes in time t are listed as 1, otherwise 0 in a 1*k matrix I_t. The spread situation can be modeled in the following step.*

$$F(P.\cdot A) = \begin{bmatrix} f(a_{11}\cdot p_{11}), f(a_{12}\cdot p_{12}), \dots \dots, f(a_{1k}\cdot p_{1k}) \\ f(a_{21}\cdot p_{21}), f(a_{22}\cdot p_{22}), \dots \dots, f(a_{2k}\cdot p_{2k}) \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ f(a_{k1}\cdot p_{k1}), f(a_{k2}\cdot p_{k2}), \dots \dots, f(a_{kk}\cdot p_{kk}) \end{bmatrix} = \begin{bmatrix} rp_{11}, rp_{12}, \dots \dots, rp_{1n} \\ rp_{21}, rp_{22}, \dots \dots, rp_{2n} \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ rp_{n1}, rp_{n2}, \dots \dots, rp_{nn} \end{bmatrix}$$

$$f(\mathrm{x}) = \begin{cases} 1 & when\ random\ number\ 0\ to\ 1 < x \\ 0 & when\ random\ number\ 0\ to\ 1 \geq x \end{cases}$$

$$\begin{aligned} I_{t+1} = I_t \cdot F(E).\cdot M &= I_t \cdot F(P.\cdot A).\cdot M \\ &= [i_{1t}, i_{2t}, \dots \dots i_{kt}\ ] \cdot \begin{bmatrix} rp_{11}, rp_{12}, \dots \dots, rp_{1n} \\ rp_{21}, rp_{22}, \dots \dots, rp_{2n} \\ \dots \dots \dots \dots \dots \dots \dots \dots \\ rp_{n1}, rp_{n2}, \dots \dots, rp_{nn} \end{bmatrix} .\cdot [m_1, m_2, \dots \dots m_k\ ] \\ &= [i_{1t+1}, i_{2t+1}, \dots \dots i_{kt+1}\ ] \end{aligned}$$

After several iteration, the whole spread matrix is shown in $I_{t->\infty}$

Referring to SIR model, we can get similar equation:

$$S_{t+1} - S_t = \sum I_t - \sum I_t \cdot F(E)$$

$$I_{t+1} - I_t = \sum I_{t+1} - \sum I_t$$

$$R_{t+1} - R_t = \sum I_t \cdot F(E) - \sum I_{t+1}$$

$$\sum F\ \ means\ the\ sum\ of\ items\ in\ matrix\ F$$

Suppressions and Preventions

After we have figured out the structure of Internet and have set up a model to analyze it, now it's time to find a best method to prevent virus from spreading on the basis to my model.

Suppressing viruses, in fact, is the race between viruses spread and patches distribution [12]. The time gap between the virus starts to spread and the patch is generated provides a window for virus to spread rampantly. In that critical time, cutting some links or disabling some part of the Internet for sacrifice is better than being infected in a much larger scale. But which edges should be cut in what order and how can we cut fewest edges to ensure safety and leave the least negative side-effect to the connectivity of Internet?

Suppression on lower layer of the Internet

First, we can have a clearer view of the hierarchy topology in the first layer in Fig. 5.

Blue nodes represent the gateway nodes in the stubs, connecting to the higher layer in the Internet, green ones are its first-generation child nodes, and grey ones the second.

The hierarchy structure is featured on its clear boundary of levels, every node has parent(s) and child(ren). So when a virus attacks a node in such a structure randomly (for example the green node), it spreads fast to its children and grandchildren nodes (its grey nodes), but slower upward to its parent node due to its degree and the fact that connections between child and parent are more likely to be child-to-parent one-way directed. However, when the virus does spread upward to its parent (the blue node), it may cause greater damage, for other green nodes are highly vulnerable. If other green nodes are infected, all their grey nodes will be infected soon.

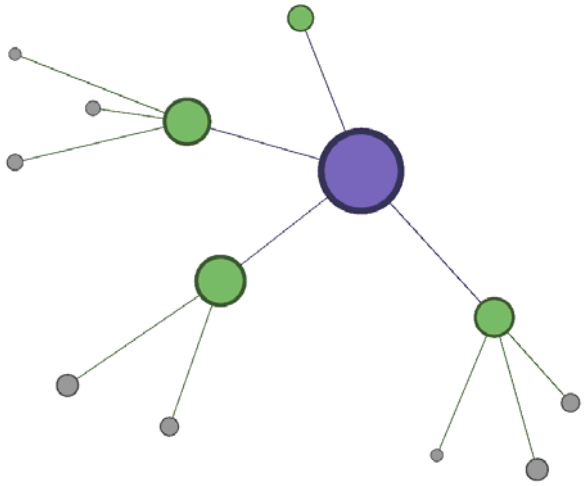


Figure 5. *Illustration of hierarchy topology in the lower network layer.*

One possible solution is to cut the connection between infected nodes and their parents as soon as possible. It cuts the fewest links but may prevent the virus from spreading to higher levels in a hierarchy structure. In comparison, those child nodes are relatively meaningless to rescue. They are large in quantity, and are more likely to have been infected before the instruction of unlinking them from their parent nodes. If there exist link-to-other-network edges or nodes in that hierarchy network, remove them as well to prevent virus from spreading to other network.

But when the virus intentionally attack the gateway node, which is highly favorable to the spread of virus, what will we do? Abandon or rescue? The GPR ratio in *Definition 7* gives us a helpful suggestion. For the spread of virus is relatively faster than our protection, so it's nearly impossible to rescue its direct child nodes, so when the hierarchy network has a high GPR, why should we bother to rescue few meaningless nodes under that structure? We can save our energy and time to protect other stub domain networks. When its GPR is relatively low (for example below 50%), which means still a lot of valuable nodes haven't been infected after the initial spread, then we should calculate every of its child nodes' GPR, find the larger ones and give them immediate protection. The protection order should follow their GPR number, protecting larger ones preferentially.

Suppression on higher layer of the Internet

While the higher level of the Internet cannot simply be described as a certain type of topology. Instead we should view the overall network in a broader perspective.

In the model of transit domain, the structure of every stub domain has been reflected in the GPR and HNII values, which means the core spirit of suppression and protection should be in

centered with how to avoid gateway nodes with high HNII and GPR being infected. Those edges between gateway nodes can be cut and some backbone nodes between gateway nodes can be removed intentionally to protect these gateway nodes.

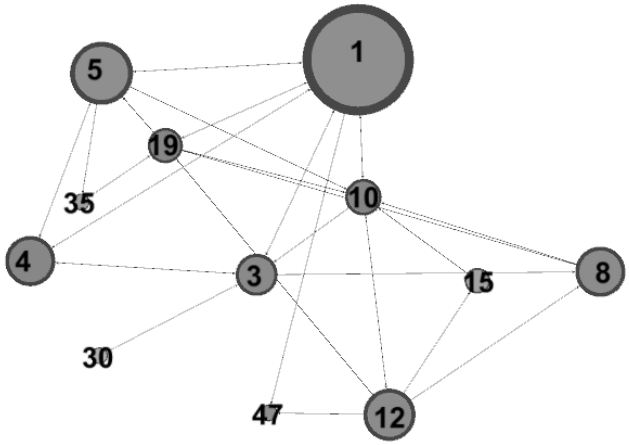


Figure 6. *An example of transit domain layer model simplified from Fig.3. The size of circle relatively shows their HNII and GPR value.*

In order to demonstrate in a typical and clear way, a few adjustment has been made on the base of Fig. 2, and I convert gateway nodes' HNII and GPR value into the circle size of nodes. We can find that node 1 is the largest (its GPR is 87.33% and HNII is 4.1195, nearly doubled to the example in the model), so we should protect node 1 on top priority.

Definition 14: The threshold of a gateway node security is determined by the GPR and HNII of gateway nodes

$$T_i = \frac{K_t}{GPR_i \cdot HNII_i}$$

K_i is the manually selected threshold to control safety level. The lowerer K_t is, the safer restriction you choose.

To avoid potential risk, we need to ensure e_i of every gateway node is below the threshold of T_i. The best cutting edge plan is to cut from the node j with $\max\{a_{ji} \cdot p_{ji}\}$ to $\min\{a_{ji} \cdot p_{ji}\}$(j is the possible nodes it connects to).

Take Fig. 6 as an example.

Connection matrix

A =

0	0	1	1	1	1	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	1	0	1
1	0	0	1	0	0	1	1	0	1	0	0
1	0	1	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	1	1	0	0
1	0	0	0	1	0	0	1	1	1	0	1
0	0	1	0	0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0	0	1	0	1
0	0	0	0	1	1	0	0	0	0	0	0

1	1	1	0	1	1	0	1	0	0	0	1
1	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	1	0	1	0	1	1	0

Infection possibility matrix(random)

P=

0.728672	0.562794	0.721597	0.706984	0.649372	0.588013	0.562977	0.832143	0.930629	0.03093	0.571356	0.053905
0.613984	0.990311	0.699716	0.694164	0.665636	0.533702	0.230883	0.660822	0.636444	0.626901	0.615856	0.787037
0.900598	0.957315	0.460747	0.265121	0.230147	0.472616	0.144762	0.587068	0.047771	0.936997	0.560914	0.630077
0.252779	0.336818	0.170593	0.326574	0.80489	0.323811	0.677096	0.019078	0.698346	0.319935	0.279859	0.508819
0.220095	0.155456	0.794371	0.093234	0.502785	0.430986	0.354071	0.83695	0.25711	0.745175	0.15185	0.567686
0.235805	0.417079	0.8885	0.041599	0.43403	0.764898	0.510472	0.113998	0.690902	0.288928	0.733013	0.309191
0.81013	0.554477	0.142515	0.75928	0.709034	0.996463	0.694707	0.84042	0.36318	0.124354	0.039699	0.747136
0.702409	0.029752	0.939672	0.324874	0.555264	0.265018	0.595293	0.337497	0.541884	0.514081	0.801622	0.336661
0.878505	0.866706	0.000261	0.521822	0.255004	0.792067	0.506111	0.429436	0.222142	0.387512	0.406417	0.883354
0.711011	0.894414	0.438805	0.951888	0.075201	0.134609	0.776399	0.793779	0.905487	0.97861	0.830011	0.333656
0.66108	0.49719	0.011168	0.668978	0.589102	0.035536	0.685203	0.141567	0.373541	0.291818	0.971035	0.796015
0.439577	0.209287	0.234516	0.473122	0.674242	0.453946	0.319494	0.608102	0.078816	0.592343	0.422483	0.804784

Each node GPR=[87.33%;0;41.31%;21.93%;23.12%;100%;100%;100%;0;100%;0;68.31%]

(0 means it is not gateway nodes but backbone nodes in transit domains)

Each node HNII=[4.1195;0;2.6287;2.7211;3.0113;0.5893;0.1344;2.8131;0;1.5821;0;3.7182]

Choose K_t=2, a relatively medium threshold.

Each node threshold T_i=[0.555926173;
∞;1.841763134;3.351558943;2.872685894;3.393857119;14.88095238;0.71095944;
∞;1.264142595;∞;0.787431826]

Before cutting edges:

b. *p	$q_{ji}*p_{ji}$											
	0	0	0.7216	0.707	0.6494	0.588	0	0	0	0.0309	0.5714	0
	0	0	0	0	0	0	0	0	0	0.6269	0	0.787
	0.9006	0	0	0.2651	0	0	0.1448	0.5871	0	0.937	0	0
	0.2528	0	0.1706	0	0.8049	0	0	0	0	0	0	0
	0.2201	0	0	0.0932	0	0.431	0	0	0.2571	0.7452	0	0
	0.2358	0	0	0	0.434	0	0	0.114	0.6909	0.2889	0	0.3092
	0	0	0.1425	0	0	0	0	0	0	0	0	0
	0	0	0.9397	0	0	0.265	0	0	0	0.5141	0	0.3367
	0	0	0	0	0.255	0.7921	0	0	0	0	0	0
	0.711	0.8944	0.4388	0	0.0752	0.1346	0	0.7938	0	0	0	0.3337
	0.6611	0	0	0	0	0	0	0	0	0	0	0.796
	0	0.2093	0	0	0	0.4539	0	0.6081	0	0.5923	0.4225	0
Ti	2.9814	1.1037	2.4132	1.0653	2.2185	2.6646	0.1448	2.103	0.948	3.7353	0.9939	2.5626
Kt/vi	0.555926	Inf	1.841763	3.351559	2.872686	3.393857	14.88095	0.710959	Inf	1.264143	Inf	0.787432

After cutting edges (edges been cut are painted as yellow):

b. *p	$q_{ji}*p_{ji}$											
1	0	0		0.707	0.6494	0.588	0	0	0	0.0309	0.5714	0
2	0	0	0	0	0	0	0	0	0		0	
3		0	0	0.2651	0	0	0.1448	0.5871	0		0	0
4		0	0.1706	0	0.8049	0	0	0	0	0	0	0
5	0.2201	0	0	0.0932	0	0.431	0	0	0.2571		0	0
6	0.2358	0	0	0	0.434	0	0	0.114	0.6909	0.2889	0	0.3092
7	0	0	0.1425	0	0	0	0	0	0	0	0	0
8	0	0		0	0	0.265	0	0	0	0.5141	0	
9	0	0	0	0	0.255	0.7921	0	0	0	0	0	0
10		0.8944	0.4388	0	0.0752	0.1346	0		0	0	0	0.3337
11		0	0	0	0	0	0	0	0	0	0	
12	0	0.2093	0	0	0	0.4539	0		0		0.4225	0
Ti	0.4559	1.1037	0.7519	1.0653	2.2185	2.6646	0.1448	0.7011	0.948	0.8339	0.9939	0.6429
kt	0.555926	Inf	1.841763	3.351559	2.872686	3.393857	14.88095	0.710959	Inf	1.264143	Inf	0.787432

Only cutting 15 edges can make sure the network is safe under threshold K_t

Simulations

Simulations are based on Fig. 2. During simulation, there may exist minor changes to better illustrate. As possibility to be infected between nodes is not certain, during simulation, the matrix

P is based on random number. Due to the limited time during research, some process in simulation is manually operated instead of program testing.

The simulation is based on the condition:

1. Infection/ immunization has steps of time. Each step an infected node attacks each of its neighbors once, and we can immunize (recover) one node once. Immunized nodes can recover its child nodes.
2. Immunization begins on the sixth step when patches have been developed.
3. Every node has a random chance to be infected when there's a connection between infected nodes. If not succeed, the attack will be continued until the target is infected.

It is a roughly presentation of virus attack, and my most concern is not recovery but whether the initial $I_{t+1} - I_t$ which means the speed of spread is slowed down effectively at the first stage, and whether the immunization is effective enough.

The 'stories' are listed as tables.

Intentional attack

Time	Susceptible	Infected	Recovered	'Story'
1	99	1	0	Intentionally attack node 1
2	78	22	0	For a high number in GPR and large degree in transit domains, almost every node 1 child infected and the transit domains suffered a lot. At the same time, virus alarm triggered, gateway nodes start to calculate their GPR and HNII and cut their highly infective edges.
3	68	32	0	Virus are spreading to gateway node 5, for it is closely connected with node 1. Gateway nodes completed cutting edges and lower their e_i under threshold. $I_{t+1} - I_t$ has fallen to 10.
4	60	40	0	Node 5 has a high GPR too, so it causes great damage to stub networks under node 5. While other gateway nodes have completed precautions, so not infected. $I_{t+1} - I_t$ has fallen to 8.
5	56	44	0	The cutting edges plan works well, no more gateway nodes infected, only some backbone nodes in transit domains infected. The great news that immunization (recovery) has been developed. $I_{t+1} - I_t$ has fallen to 4.
6	53	47	0	Immunize node 12, which has a high GPR and HNII, and surrounded by many infected nodes. $I_{t+1} - I_t$ has fallen to 3. The reason not choose recover node 1 is because its high GPR means nearly all its child node has been infected, so abandon that stub and save others.
7	50	50	0	Immunize node 8, which also has a high GPR and relatively high HNII. $I_{t+1} - I_t$ has remains as 3.

8	48	47	5	Immunize node 4, which has a medium HNII. A number of nodes has been recovered
9	47	40	13	Major gateway nodes has been immunized, so it's time to recover. Recover node 1, which has the largest degree. The $R_{t+1} - R_t$ is boosting. $S_t - S_{t+1}$ has fallen to 1.
10	46	26	28	Recover node 5.$S_t - S_{t+1}$ is 0 since then on. We have saved 46 nodes at last, which mostly under gateway nodes 12,8,4.
11	46	15	39	Recover node 3.
12	46	10	44	Recover node 2.
13	46	5	49	Recover node 14.
14	46	3	51	Recover node 91.
15	46	0	54	All nodes are healthy.

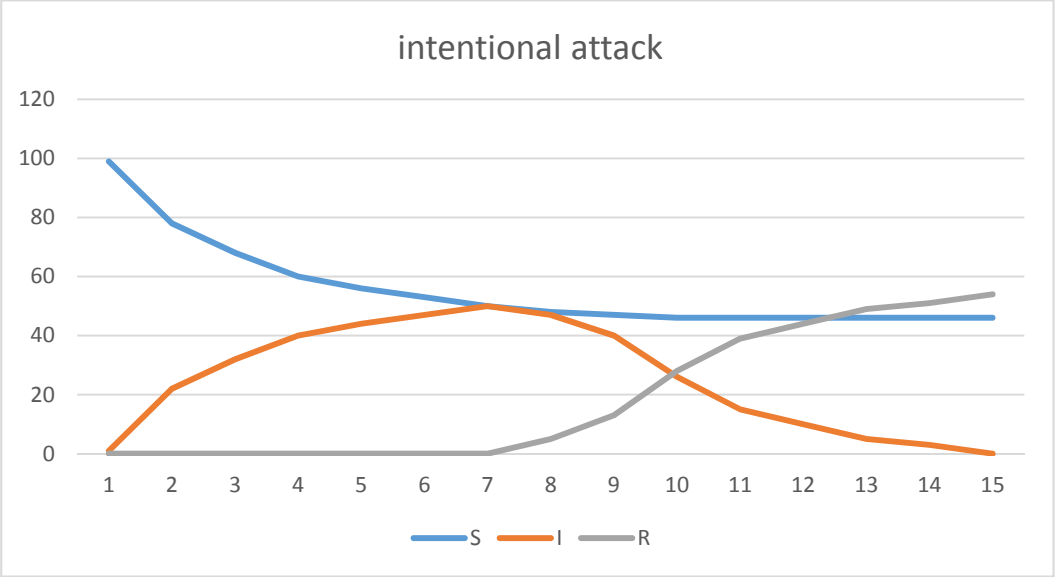


Figure 7. *The SIR number in chart during an intentional attack.*

Random attack in transit domains

Time	Susceptible	Infected	Recovered	'Story'
1	100	1	0	Random attack node 23
2	96	4	0	For a medium degree node in the middle of the transit domain, begin to attack gateway node 1. Virus spread to node 3, which is an important node connect to three important gateway nodes. At the same time, virus alarm triggered, gateway nodes start to calculate their GPR and HNII and cut their highly infective edges.
3	87	13	0	Node 1 cut its edge with node 23 and 3, avoid direct attack. Other gateway nodes do similar prevention.

				Node 10 and 12, however, facing many virus neighbor, still fail to prevent. $I_{t+1} - I_t$ has risen to 9.
4	77	23	0	Nodes under 10 and 12 are largely infected, but other gateway nodes are resisting virus. $I_{t+1} - I_t$ has risen to its peak 10.
5	71	29	0	The cutting edges plan works well, no more gateway nodes infected, but the majority of nodes in transit domains are infected. The great news that immunization (recovery) has been developed. $I_{t+1} - I_t$ has fallen to 6.
6	67	33	0	Immunize node 1, which has a top degree, HNII and high GPR, and surrounded by many infected nodes. $I_{t+1} - I_t$ keeps falling.
7	64	35	1	Immunize node 5, which also has a high GPR and high HNII.
8	61	35	4	Immunize node 8, which has a medium HNII. A number of nodes has been recovered.
9	60	32	8	Immunize node 4, which has a medium GPR and HNII.
10	59	28	13	Major gateway nodes has been immunized, so it's time to recover. Recover node 12. The $R_{t+1} - R_t$ is boosting. $S_t - S_{t+1}$ has fallen to 1.
11	59	18	23	Recover node 5. $I_{t+1} - I_t$ is 0 since then on. We have saved 59 nodes at last, which mostly under gateway nodes 1,5,4,8.
12	59	6	35	Recover node 98
13	59	3	38	Recover node 80
14	59	0	41	Recover node 44

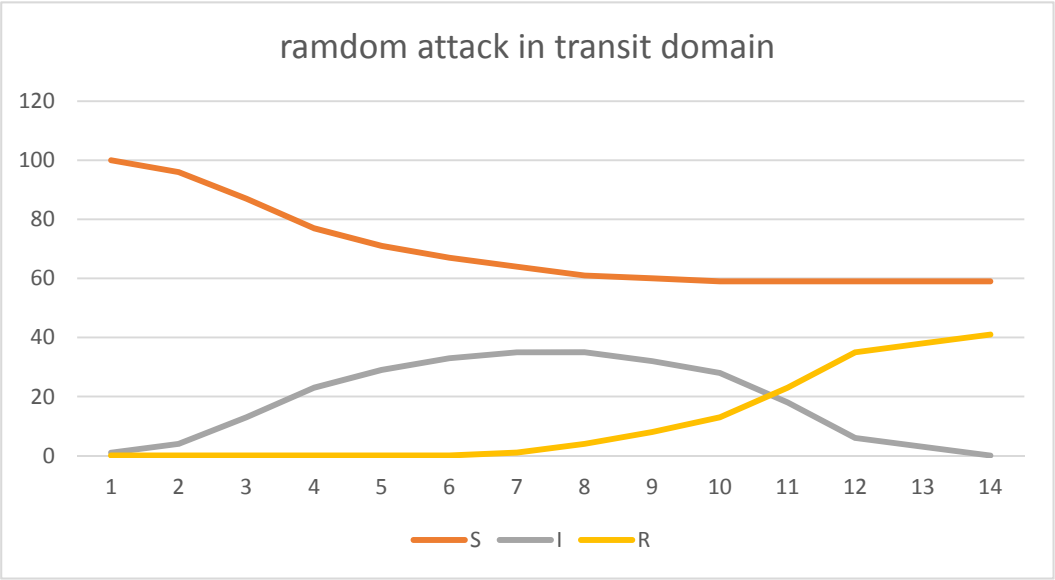


Figure 8. *The SIR number in chart during a random attack in transit domains.*

Random attack in stub domains

Time	Susceptible	Infected	Recovered	'Story'
1	100	1	0	Random attack node 83, which under gateway node 1.
2	97	3	0	It spread fast to node 85 and 7, which are its child node. At the same time, virus alarm triggered under gateway 1, node 1 cuts its edge to node 83. And it removes node 9, which indirectly connects node 83 and node 1.
3	95	5	0	Viruses spread deeper to node 63 and 92.
4	95	5	0	Spread is isolated in that tiny area, while other stub networks and transit networks remains the same.
5	95	5	0	
6	95	4	1	Recover node 83
7	95	2	3	Recover node 55
8	95	0	5	Recover node 63. Reconnet node 9.

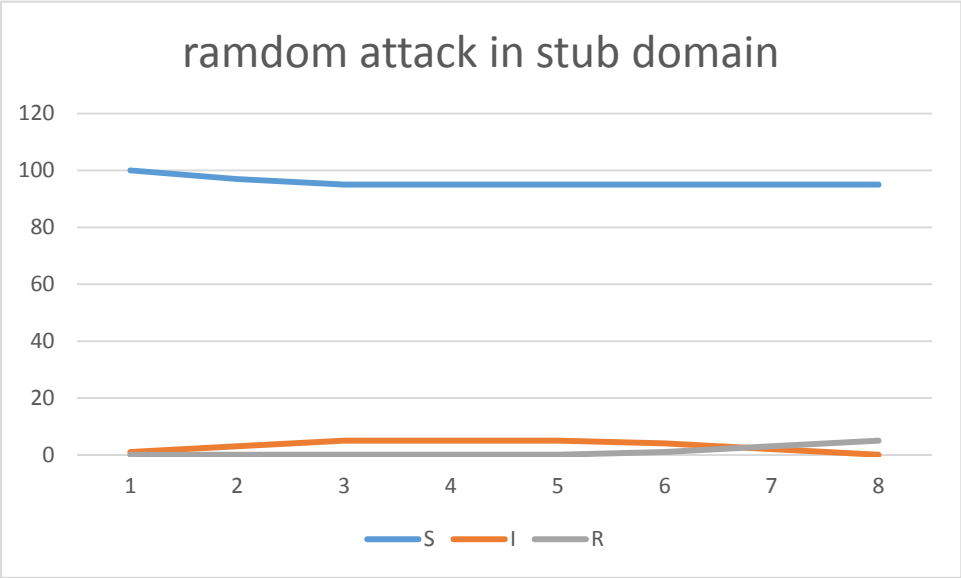


Figure 9. *The SIR number in chart during a random attack in stub domains.*

Conclusion

Since the network diameter is 6, in 6 steps of prevalent spreading, such a network is sure to be all infected if no protection is taken. While in my protection plan, cutting relatively fewest connections ensures the greatest possible safety. From these three simulations, we can find the result is pleasant. Especially in the time 3 to 6, when immunization is not developed, it surely reduce the $I_{t+1} - I_t$ numbers. Data reveals that the $I_{t+1} - I_t$ begins to decrease after cutting edge process is finished. While under no protection, $I_{t+1} - I_t$ will continue to boost until most nodes are infected.

But, we can also find that the situation is largely different according to the attack type.

The intentionally attack is highly vicious. It destroys the largest degree nodes, so it spreads fast in transit domains; it has large GPR, so it spreads in stub domain fast; it has large HNII, so it can efficiently affect large numbers of vulnerable and valuable nodes. While taking my plan to suppress virus, it has effectively ensured the safety of 46 nodes, which mostly under the protected gateway nodes, meaning gateway nodes play an essential role in Internet.

Random attack is better to control, even the random node has a degree of 7 and one of them is the largest degree node, finally we still save 59 nodes.

Random attack in a stub network is the most controllable. Owing to its hierarchy structure, after cutting the vicious part out of the whole network, it can no longer spread out. In my simulation, 95% of the nodes are not infected and only 3 edges and 1 node are removed, the rest part of Internet remains healthy and functional.

Virus spread is a very complicated process, there are many different aspects should be considered into model and there are many random occasions that are not easily revealed by numbers and equations. My model and assumption have taken geometric factor, spread possibility, network structure into consideration, calculating revelatory indexes GPR and HNII to describe the stub domain networks and using e_i to value spread possibility between transit domain nodes. However, there might still be many factors to consider in the real world. All the simulation and result only work in my model in a strictly confined perspective.

All the above is my own theory on virus spread, intriguing more people who are interested in complicated network to continue the research. There is still a long way to go to perfectly describe the virus spreading situation and find more effective ways to suppress and prevent virus from spreading.

Reference

1. Watts, D. J. & Strogatz, S. H. Collective dynamics of ‘small-world’ networks.Nature 393, 440–442 (1998).

2. Baraba´si, A. L.&Albert, R. Emergence of scaling in random networks. Science 286,5439, 509–512 (1999).

3. Erdo¨s, P. & Re´nyi, A. On the evolution of random graphs. Publications of the Mathematical Institute of the Hungarian Academy of Sciences 5, 17–61 (1960).

4. X. Li. Complicated dynamic network propagation mechanics. Proceeding on mechanics. 2008(06).

5. Gouhei Tanaka Kai Morino & Kazuyuki Aihara. Dynamical robustness in complex networks: the crucial role of low-degree nodes. SCIENTIFIC REPORTS , 2 : 232.

6. R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin,Phys. Rev. Lett. 85, 4626 (2000).

7. R. Albert, H. Jeong, and A.-L. Barab´asi, Nature 406,378 (2000).

8. Kenneth L. Calvert, Matthew B. Doar, Ellen W. Zegura. Modeling Internet Topology. Global Telecommunications Conference, 1996. GLOBECOM '96. 'Communications: The Key to Global Prosperity, 86 – 93.

9. X. Li and G. Chen. A local-world evolving network model. Physica A, 328:274–286, 2003.

10.S. Bar, M. Gonen, and A. Wool. A geographic directed preferential Internet topology model. Technical report, 2005. arXiv:cs.NI/0502061.

11. Kermack, W. O.; McKendrick, A. G. (1927). A Contribution to the Mathematical Theory of Epidemics. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 115 (772): 700.

12. YuanYuan, Fan. Epidemic Model of Mobile Phone Virus for Hybrid Spread Mode with Preventive Immunity and Mutation.Epidemic Model of Mobile Phone Virus for Hybrid Spread Mode with Preventive Immunity and Mutation [1-4244-3708-3; 1-4244-3709-1]yr:2010 pg:1 -5.