# Modelling Spatio-Temporal Progression of Cyberattacks in Data Networks

Malik Shahzad Kaleem Awan1, Vince Knight2, Omer Rana1 and Pete Burnap1
1School of Computer Science and Informatics, 2School of Mathematics, Cardiff University, UK

***Abstract*—University data networks comprise numerous devices connected to the network on an ad-hoc basis. Measuring the risk of cyberattacks and identifying the most recent modus-operandi of cyber criminals on such networks can be difficult due to the wide range of services and applications running within the network, the multiple vulnerabilities associated with each application, the severity associated with each vulnerability, and the ever-changing attack vector of cyber criminals. We propose a spatio-temporal modelling framework to represent these features, enabling real-time network enumeration and traffic analysis, to produce quantified measures of risk at specific points in time. We validate the approach using data from a University network, with a data collection consisting of 460K instances representing threats measured over a 144 hour period. Our analysis can be generalise to a variety of other contexts.**

## I. Introduction

An institutional computer network is often divided into administrative sub-networks (and zones) to enable a systems/ network administrator to manage data traffic, software upgrades and access control policies at a finer level of granularity. However, such subnets are often not fully physically isolated from each other – with gateway machines that act as an interface between two or more subnets. While methods have been proposed for continous monitoring of networks risk [11], there is a lack of evidence on the ability to model cyberattack propagation across the network over time. This is important because different subnets will likely have varying value within an organisation. For instance, if the subnet containing low value assets is compromised it will have a much lower impact on the organisation than a subnet with personally identifiable information, compared to a subnet containing crictical assets such as safety control systems. Thus it is essential for systems/network administrators to have evidence on how far network intrusions have propagated, the likely effect on the attack on individual machines within a subnet, and potential impact on the network as a whole.

We employ a Markovian approach to modeling how a network intrusion can propagate over time to infect machines in multiple subnets. Using data acquired from an Intrusion Detection/Prevention Systems (IDS/IPS) our contributions come in the form of two models: (i) an agent-based model, which provides a fine-grained analysis of the potential behaviour of each machine within a subnet and enables a measurement of impact on that machine; and (ii) a network-level model that captures the interaction between machines within subnets following network intrusion (e.g. malware execution), measuring propagation across subnets. We can capture the software types that can be hosted on each individual machine, how this is affected by network intrusion, and how the machine eventually recovers (e.g after malware removal). The agent-based model is used to model how individual 'infected' machines support propagation to other machines in the subnet and, using a stochastic model as opposed to a deterministic *fixed* model based on assumptions relating to the underlying characteristics of the computer network, we are able to consider more dynamic and flexible forms of network intrusion and its subsequent impact. The proposed approach shows how epidemiological infection propagation models in animals can also be applicable to computer networks.

We consider a computer network to be made up of identifiable zones (subnets) – each zone being vulnerable to a cyber attack such as network intrusion and/or malware execution, with different levels of susceptibility to be attacked based on software configurations. This work can be used to provide evidence on how an attack propagates across zones at different rates and to attribute causality. The outcome can be used to: (i) harden security in the subnet; and (ii) isolate infected subnet, and the effect of these actions on the rest of the network over time. We validate our approach using data acquired from a Palo Alto University IDS/IPS system.

## II. Related Work

The ability to quantify infection of hosts starting from a single infected host has been studied previously to better understand the underlying topological characteristics of the propagation of malware in the form of a Worm. Probabilistic modelling methods and sequential growth models were used for mathematical analysis and simulation to characterize the Worm spread behaviour used by botnets. The results show that the generation follows a Poisson distribution with average

path length of infection tree increasing logarithmically with respect to total infected hosts and are used for evaluating bot assessment strategies for forensic analysis of a worm tree [1]. The study focuses on understanding the infection spread by a single worm using simulations with a set of defined properties for worm behaviour and propagation. Our models consider multiple forms of malware targeting a network at a given time and spreading the infection within different zones of a large computer network.

An extensive study on vehicular malware epidemics have been performed to determine the dynamicity in malware spread in vehicle-to-vehicle (V2V) communication due to factors such as vulnerability diffusion, ratio of penetration, V2V communication technology range and Worm self-propagation mechanism. A numerical model has been shown to mimic the worm spreading process effectively. Two synthetic datasets: Canton of Zurich and Madrid, were used for developing a dedicated simulator for large-scale vehicular networks to validate these analytical models [3].

A queue-based modelling approach for capturing Susceptible-Infected-Susceptible (SIS)-type malware has been proposed for capturing malware dynamics in time-varying networks. An evaluation of the model for exploiting analytically quantified network reliability was demonstrated, while the model has also been used for enhancing network robustness against malware attacks [4]. This study assumed the structure of the underlying network along with the propagation of malware in networks with churn, whereas our work uses real-world traffic logs to provide an actual trace for malware propagation.

A graph-theory inspired belief propagation framework has been proposed for detecting early-stage Advanced Persistent Threats (APTs) targeting and infecting a large enterprise. The framework was validated using two large datasets and reported performance levels for APT detection better than state-of-the-art security products [5]. The study has mined large data sets for early stage detection of infections in large networks. Our work differs in scope as we use markov models to understand and model the propagation of malware spread in network (after infection) to enable a network administrator to take appropriate responses.

Discrete-state Markov modeling has previously been used to analyse the propagation of malicious code in computer networks. The model uses susceptible (S), quarantine (Q), infection (I) and health (H) states for modelling the transition probability between these states [6]. The study has used experience and tacit knowledge for modelling the spread of malware whereas we have based our study on real dataset for malware spread.

A diurnal model has been proposed to understand the effects of time and location on dynamics of malware spread following botnet activities in online vulnerable populations and accordingly allocate resources in an intelligent way to apply appropriate security measures [7]. The model was evaluated using empirical data collected over a period of six months and contained dozens of botnets targeting millions of victims. The model considered time zones and time of day for modelling the growth of botnets. Rather than growth, we have used Markov modelling for analysing the propagation of network intrusions to consider various what-if scenarios with reagrd to mitigation strategies.

## III. Framework

The model used in this work is an $Z$ dimensional Markov chain where $Z$ is the number of sub networks. The state space $S$ is given by (1):

$$S = \left\{ (n_1, \ldots, n_Z\} \in \prod_{i=1}^{Z} \{0, 1, \ldots, N_i\} \right\} \quad (1)$$

Thus $S$ is defined by $(n_1, \ldots, n_Z)$ – the number of machines in zone $i$, $1 \leq i \leq Z$. The size of the state space is thus given by $|S| = \prod_{i=1}^{Z}(n_i + 1)$. For any two states $u, v \in S$ we have the transition rate between states given by $q(u, v)$ in equation (2):

$$q(u,v) = \begin{cases} \sum_{i=1}^{N} \gamma_{ij} u_i + \gamma_j^*, & \delta(u,v) = e_j \\ u_i \mu_i, & \delta(u,v) = -e_j \\ -\sum_{w \neq u} q(u,w), & \text{if } u = v \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$e_j$ is a vector with 0s in all positions except $j$ where it has value 1; $\gamma_{ij}$ denotes a rate, measuring the number of machines in subnet $j$ infected by a single machine in subnet $i$; $\gamma_j^*$ denotes a rate, measuring how a single machine from subnet $j$ is infected from outside of the network; $\frac{1}{\mu_i}$ denotes the mean length of time any machine of subnet $i$ is infected. Figure 1 shows potential states and transitions as an example.

$$Z = (2,2),\ \gamma = \begin{pmatrix} 5 & 0.3 \\ 0 & 2 \end{pmatrix},\ \gamma^* = (1,2),\ \mu = (4,4) \quad (3)$$

For a given $Q$ and initial state vector $\pi$ the probability of being in each state at given time $t$ can then be computed as

$$\pi(t) = \pi e^{Qt} \quad (4)$$

Equation (4) can be used to obtain the evolution of the expected number of infected machines in each zone over time as shown in Figure 2. For a network with 6 subnets with 100 machines per subnet, we would have $101^6 \geq$

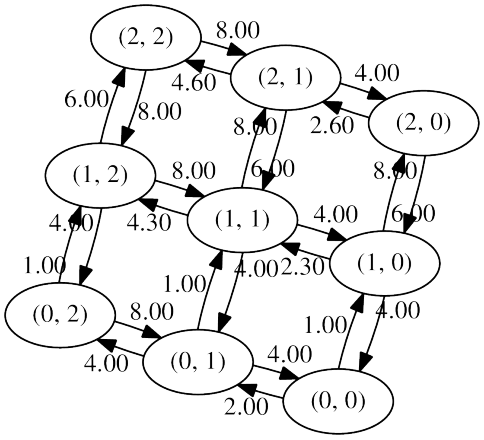


Fig. 1. States and transitions for the example of (3)

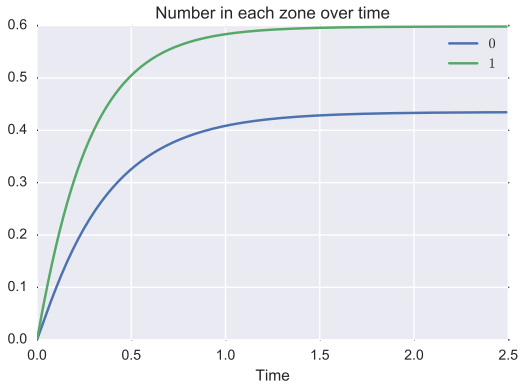


Fig. 2. Mean number of infected machines in each zone over time for example of (3)

$10^{13}$ states. Of these states it is likely that only a small proportion would be occupied for a significant proportion of the process. We therefore simulate the chain in an exploratory manner as described in Algorithm 1.

**Algorithm 1:** Exploratory simulation

```
initial state for given amount of time do
    sample time till transition to all potential states
    go to state that transition happens earliest
    if state is visited for first time then
        augment hash table with state
    end
    record time spent in previous state
end
```

Figure 3 shows the long run simulation and analytical probability distribution results for the example of (3). This plot matches the analytical results which serves as a validation of the simulation model. This framework allows for the efficient modelling of a variety of scenarios.

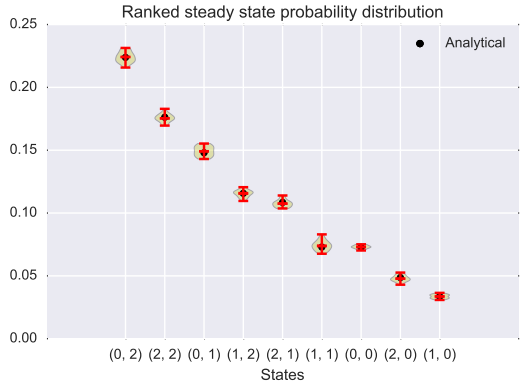


Fig. 3. Probability distributions using Algorithm 1 for example in (3). Simulating 500 time units, warm up of 200 units with 20 trials.

## IV. Validation Using an Intrusion Dection System Data Set

We have used the University network as a validation environment for our proposed framework. The University network has 34000+ registered users and is protected by Palo Alto Networks IPS/IDS software Wildfire solution. The University network has six subnets: Campus, Campus-SrvNets, DC-DMZ, Reslan, Untrust, Wireless; which represent different services offered to various stakeholders. Table 1 gives a description of the six subnetworks.

If incoming traffic instance, via any subnet, is classified as malicious by Wildfire, then that traffic instance is pushed into a RabbitMQ queuing system, which is setup for storing and analysing the malicious traffic data. Each malicious traffic instance is represented by 41 attributes, which give information about both source and destination ip addresses, ports, zones and countries; threats, threat categories, threat severity levels, threat occurrence time; software applications targeted; classification rule, protocols, ingress and egress interfaces as well as miscellaneous information.

We selected malicious traffic logs for 6 consecutive days over an academic year: March 26-31, 2014. The accessed logs were in CSV format and represented 462,787 instances with 278 unique threats belonging to 6 distinct threat categories: vulnerability, file, virus, scan, spyware and packet. These threats targeted 90 different software applications with various levels of severity. The Wildfire solution uses 5 labels to represent severity levels: informational, low, medium, high and critical. The log files were pre-processed to extract information about threats, number of infected hosts in source subnet, infected hosted in destination subnet and occurrence time of malicious traffic incidents, which were then grouped on an hourly basis in terms of threat occurrence time.

TABLE I
SUBNETWORKS IN THE UNIVERSITY NETWORK

| Subnetwork | Description |
|---|---|
| Campus | This subnet contains workstations mainly used by staff and in classrooms/labs |
| Campus-SrvNets | This subnet provides processing of non-sensitive information as part of campus services |
| DC-DMZ | This subnet has physical servers providing front end/middleware services to trusted backends and is likely to be exposed to the Internet |
| Reslan | This subnet contains the computing devices used in student residences network |
| Untrust | This subnet provides the network connection routes to all sponsored sites as well as the Internet |
| Wireless | This subnet provides current guest roaming and extended eduroam services |

The extracted information was then used for calculating the infection rate from a source subnet to a destination subnet during a selected time window. Each of the subnet has distinct security policies enforced by the network administrator to protect it from different types of cyberattacks.

We consider a spatiotemporal progression of threats targeting different subnets, exploiting loopholes in security policies enforced at different subnets. In [11], we proposed a dynamic probabilistic risk assessment model for calculating risk score associated with various threats targeting applications. The calculated risk score was used to identify risk hotspots posed by internally and externally hosted software applications, to help an administrator determine and understand the most *at-risk* components within a network, sub-network and hosted application over time.

This work identifies how threats entering any subnet can spread across the whole network, to help a network administrator take appropriate security measures. Monitoring of each individual machine for infection spread requires significant computational and human resources, thus, we simplified the case of infection spread by assuming that each instance of the threat infected a distinct machine. However, a machine can spread 2 or more threats to multiple and distinct machines (in the same or another subnet).

Considering our dataset, for example at time t=6, two threats: a) SSH User Authentication Brute-force attempt and b) Suspicious DNS Query; from outside of the University network entered Campus subnetwork. Each threat entering the network infects one distinct machine respectively in the Campus subnetwork. This resulted in 2 machines infected in the Campus subnetwork due to 2 distinct threats. Thus, at time t=6 the rate of machines getting from outside of the network, $\gamma^*$, is a count of infected machines in a considered subnetwork. This gives a value of $\gamma^*_{Campus} = 2.0$. We used $\gamma_{ij} = \frac{j_{infected}}{i_{infected}}$ for calculating infection rate where $j_{infected}$ represents the number of machines infected in the destination subnet 'j' originating from source subnet 'i'; and $i_{infected}$ represents the number of infected machines present in source subnet 'i'.

The 2 infected machines in the Campus subnet then became the source for spreading these particular threats and targeted distinct machines in Campus-Srvnets and Untrust. At t=6 there was only one infected machine due to the threat SSH User Authentication Brute-force attempt coming from outside of the University network into Campus while it infected 12 distinct machines in Campus-Srvnets. This resulted in an infection spread rate of $\gamma(Campus, Campus - Srvnets) = 12$ from Campus to Campus-Srvnets. Similarly, there was only one infected host in Campus due to Suspicious DNS Query that was the source of infection for 34 machines in Untrust subnet. This resulted in an infection spread rate of $\gamma(Campus, Untrust) = 34$ from Campus to Untrust. Similarly, the infection spread rates were calculated for the considered time window of 144 hours between different subnetworks.

We modelled the infection spread rate, $\gamma_{ij}$, from an infected machine located in a particular subnet to other connected subnet during the considered time window of 144 hours using a directed graph (see Figure 5). Our dataset showed Untrust subnet spreading infection to all other subnet connected to it. The infection spread rates from Untrust to Campus, Campus-SrvNets, DC-DMZ, Reslan and Wireless were $\gamma(Untrust, Campus) = 485.95$, $\gamma(Untrust, Campus - Srvnets) = 34.56$, $\gamma(Untrust, DC - DMZ) = 7.70$ $\gamma(Untrust, Reslan) = 22.50$ and $\gamma(Untrust, Wireless) = 8.39$. Considering the case of infected machines located in Campus subnet, the infection spread rates from Campus to Campus-SrvNets, DC-DMZ and Untrust subnets were found to be $\gamma(Campus, Campus - Srvnets) = 1.52$, $\gamma(Campus, DC - DMZ) = 0.01$ and $\gamma(Campus, Untrust) = 2.46$. Reslan subnet was found to be the least source of infection spread rate and infected DC-DMZ and Untrust subnets at infection spread rates of $\gamma(Reslan, DC - DMZ) = 0.03$ and $\gamma(Reslan, Untrust) = 0.02$. The machines in DC-DMZ subnet showed a self-infection pattern at a rate of $\gamma(DC - DMZ, DC - DMZ) = 2.35$ while it also lead to infection spread in Untrust subnet at a rate of $\gamma(DC - DMZ, Untrust) = 2.06$.

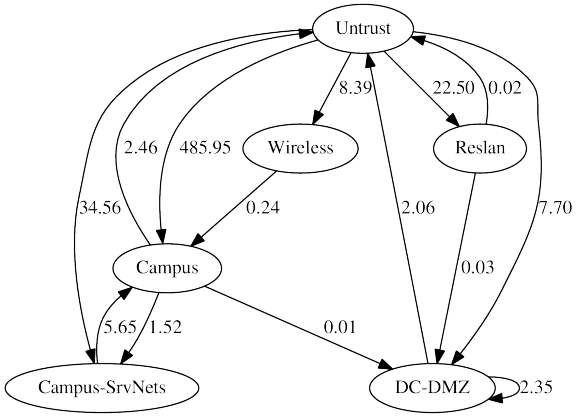


Fig. 4. The subnetworks and corresponding infection rates considered

The infected machines from both Campus-SrvNets and Wireless targeted the machines in the Campus subnet with an infection rate of $\gamma(Campus - Srvnets, Campus) = 5.65$ and $\gamma(Wireless, Campus) = 0.24$ respectively.

The Markov model proposed in Section 3 was applied for getting a better understanding of the spatio-temporal progression of malwares in a large computer network. Our framework can be generalised for malicious data logs from other commonly used network security software such as from McAfee, Symantec, Sophos and Zone Alarm.

### A. Results

A summary of data from the Wildfire IDS/IPS from Palo Alto within the University can be found in Table II.

TABLE II
SUMMARY OF DATA

| | Source Infection | Destination Infection | Infection Rate |
|---|---|---|---|
| count | 93.0 | 93.0 | 93.000000 |
| mean | 2861.47 | 1869.76 | 126.365776 |
| std | 4871.68 | 3385.90 | 962.48 |
| min | 1.0 | 1.0 | 0.000079 |
| 25% | 34.0 | 16.0 | 0.015 |
| 50% | 151.0 | 168.0 | 3.48 |
| 75% | 5214.0 | 2232.0 | 26.59 |
| max | 21246.0 | 21246.0 | 9277.0 |

Using the data obtained in Section IV and shown in Figure 4 we have the transitions between the states given by $\gamma$ in 5.

$$\gamma = \begin{pmatrix} 0 & 8.39 & 485.95 & 34.56 & 7.70 & 22.50 \\ 0 & 0 & 0.24 & 0 & 0 & 0 \\ 2.46 & 0 & 0 & 1.52 & 0.01 & 0 \\ 0 & 0 & 5.65 & 0 & 0 & 0 \\ 2.06 & 0 & 0 & 0 & 2.35 & 0 \\ 0.02 & 0 & 0 & 0 & 0.03 & 0 \end{pmatrix}, \tag{5}$$

This implies an ordering of states of the Markov model corresponding to zones of the network, to provide a baseline scenario. We consider the following number of machines in each zone is assumed: (i) Untrust (10 machines); (ii) Wireless (500 machines); (iii) Campus (200 machines); (iv) Campus-SrvNets (15 machines); (v) DC-DMZ (5 machines); (vi) Reslan (15 machines). For the purposes of the baseline scenario it is furthermore assumed that: (a) all machines in the network recover at a rate of 4 machines per time unit; (b) Untrust, Wireless and Campus subnets become infected **from outside of the network** at a rate of 1, 0.25, and 0.5 machines per time unit (respectively).

As described in Section III, the base parameters used are given in (6-7).

$$Z = (10, 500, 200, 15, 5, 15), \tag{6}$$

$$\gamma^* = (1, 0.25, 0.5, 0, 0, 0),\ \mu = (4, 4, 4, 4, 4, 4) \tag{7}$$

Using Algorithm 1 it is possible to then simulate the Markov process, Figure 5 shows the evolution of this system over 3 time units (and repeated for 20 trials). It is evident that the choice of parameters shows that all subnets have a high number of infections.

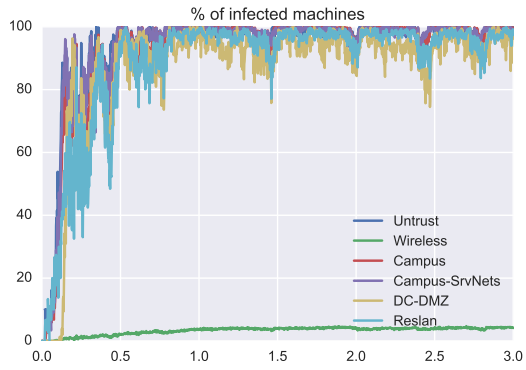


Fig. 5. Evolution of parameters for base system.

Table III shows change in infection for different sub time periods for each network: $\Delta_{t_1,t_2}$ corresponds to the change in number of infected machines between time

point $t_1$ and $t_2$. It is evident that after a very sharp rise in the number of infections, there is little change. This is despite the fact that outside infections occur at a very small rate per machine (7) and is due to the number of individual machines which, once infected are in effect weaponized. Faced with this high infection percentage; a network manager can: (a) Isolate certain subnets; (b) Improve the overall recovery rate of all machines.

TABLE III
CHANGE OF INFECTION NUMBERS OVER TIME FOR THE BASE PARAMETERS.

| Network | Untrust | Wireless | Campus | Campus-SrvNets | DC-DMZ | Reslan |
|---|---|---|---|---|---|---|
| $\Delta_{0,0}$ | 9.87 | 9.67 | 199.62 | 14.97 | 4.95 | 13.59 |
| $\Delta_{0.5,1}$ | 0.13 | 8.72 | -0.15 | -0.1 | 0.02 | 1.27 |
| $\Delta_{1,2}$ | -0.16 | 1.41 | 0.06 | -0.02 | -0.13 | -0.36 |
| $\Delta_{1.5,2}$ | -0.36 | 0.92 | -13.14 | -0.71 | -0.61 | -0.77 |
| $\Delta_{2,2}$ | 0.51 | 0.75 | 13.14 | 0.69 | 0.56 | 0.96 |
| $\Delta_{2.5,3}$ | -0.01 | -0.95 | -0.08 | 0.03 | 0.03 | -0.06 |

**Isolating subnetworks:** Figure 6 shows the effect of isolating the Untrust subnet from the Campus subnet network. Note that this isolation occurs after the 3rd time unit. Although the Campus subnet does benefit from this isolation, but the rest of the network continues to be highly infected.

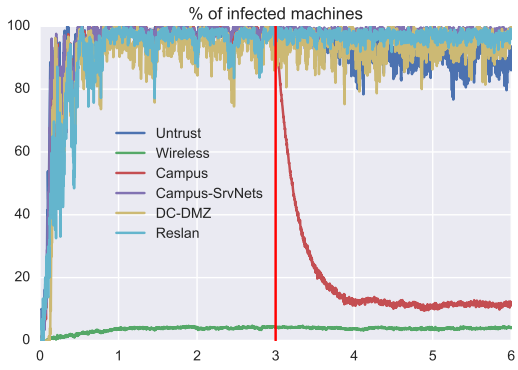


Fig. 6. Isolating campus from Untrust subnet after 3 time periods.

Figure 7 shows the effect of isolating the Campus as well as the the Reslan network from the Untrust network.

As before this only seems to have a beneficial effect on the isolated networks. Figure 8 shows the effect of completely isolating the Untrust network. Table IV shows the change of infection numbers for different sub time periods for each network. There remains a large variability with some subnets returning to a high infection rate. This is due to the machines in other networks still being able to cause spikes in infections.

**Recovery of machines:** Recalling (7), in our base scenario all machines recover at a rate of 4 per time unit. Figure 9 shows the evolution of this system after

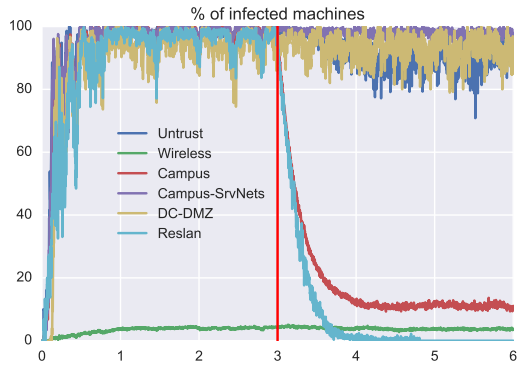


Fig. 7. Isolating Campus and Reslan from Untrust subnet after 3 time periods.

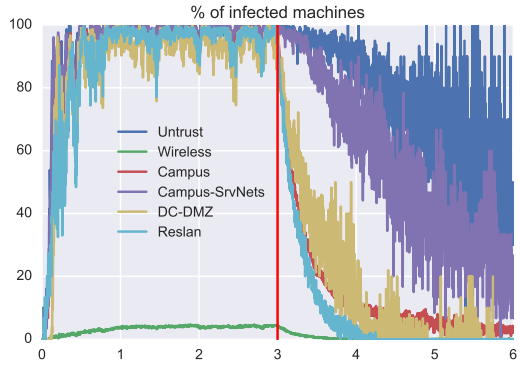


Fig. 8. Completely isolating the Untrust network after 3 time periods.

TABLE IV
COMPLETELY ISOLATING THE UNTRUST NETWORK.

| Network | Untrust | Wireless | Campus | Campus-SrvNets | DC-DMZ | Reslan |
|---|---|---|---|---|---|---|
| $\Delta_{3,4}$ | -0.31 | -18.2 | -154.7 | -1.36 | -2.99 | -12.36 |
| $\Delta_{3.5,4}$ | -0.73 | -2.71 | -22.36 | -3.16 | -1.05 | -1.95 |
| $\Delta_{4,4}$ | -3.14 | -0.14 | -10.29 | -4.0 | -0.71 | -0.29 |
| $\Delta_{4.5,5}$ | 2.62 | 0.41 | -0.11 | 0.46 | 0.0 | 0.0 |
| $\Delta_{5,6}$ | -1.33 | -0.56 | -3.89 | -2.22 | 0.33 | 0.0 |
| $\Delta_{5.5,6}$ | -4.0 | 0.0 | -4.0 | -3.67 | -0.33 | 0.0 |

reducing the infection time by 3 ($\mu_1 = 12$ for Untrust (corresponding to improving the reaction to infection).

However improving the recovery rate for this network we see a much larger effect. Figure 10 involves reducing infection time by 3 ($\mu_1 = 400$ for Untrust subnet Table V shows the change of infection numbers for different sub time periods for each network. Isolating a subnet does not inherently remove malware from the network and it may spread via other routes. However, by improving the identification and resolution of malware in the Untrust subnet we are able to stop the malware from spreading. The drastic improvement of infections for machines in the Untrust network would correspond to monitoring and

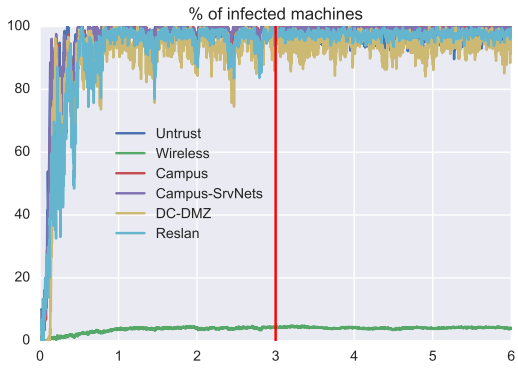


Fig. 9. Evolution of system after changing $\mu_1$ to 12.

taking remedial action.

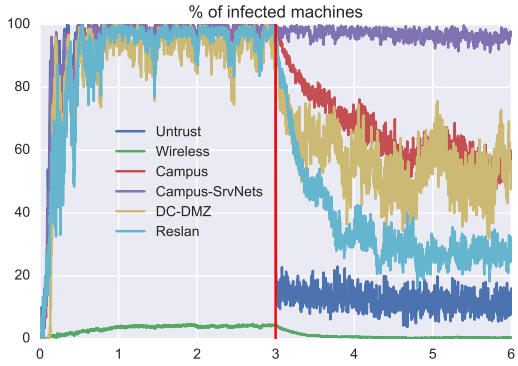


Fig. 10. Evolution of system after changing $\mu_1$ to 400.

TABLE V
CHANGE OF INFECTION NUMBERS AFTER $\mu_1 : 4 \rightarrow 400$.

| Network | Untrust | Wireless | Campus | Campus-SrvNets | DC-DMZ | Reslan |
|---|---|---|---|---|---|---|
| $\Delta_{3,4}$ | -8.05 | -16.16 | -42.48 | -0.02 | -1.48 | -7.14 |
| $\Delta_{3.5,4}$ | -0.46 | -2.15 | -21.26 | -0.33 | -0.15 | -2.87 |
| $\Delta_{4,4}$ | -0.09 | -0.72 | -24.69 | -0.04 | -0.33 | -0.31 |
| $\Delta_{4.5,5}$ | -0.15 | -0.61 | 4.84 | -0.11 | -0.15 | -0.32 |
| $\Delta_{5,6}$ | 0.03 | 0.3 | 0.29 | 0.21 | 0.37 | 0.15 |
| $\Delta_{5.5,6}$ | 0.42 | -0.13 | -3.8 | -0.08 | -0.31 | 0.68 |

## V. CONCLUSION

A framework to model and assess the spread of cyber threats in a network with multiple subnets, using a dynamic probabilistic risk assessment model, is described. The framework focuses on identifying the risk of infection across different subnets in a University network – demonstrated through six subnets using the Palo Alto Networks Wildfire IPS/IDS for threat detection. Malicious traffic is logged over six days, detecting threats across 90 applications with varying severity levels. The infection rates between subnets were calculated, using an hourly breakdown of threat occurrences. A simplified model assumes each threat infects one distinct machine, although a machine can transmit multiple threats to others within or across subnets.

Simulation results demonstrate how threats spread and validate the model through long-term simulation of infection rates across subnets. The analysis shows that isolating subnets or improving recovery rates can reduce infection rates, although an isolated subnet's benefit is often limited to itself. The work describes the use of spatiotemporal threat progression and highlights the need for network administrators to take preemptive measures against threat propagation, aiming to improve overall network security.

**Acknowledgement:** This work was supported by the Engineering and Physical Sciences Research Council (CEReS) programme grant number: EP/K03345X/1.

## REFERENCES

[1] Q. Wang, Z. Chen and C. Chen, On the Characteristics of the Worm Infection Family Tree, IEEE Transactions on Information Forensics and Security, vol. 7, no. 5, October 2012, pp. 1614-1627.
[2] W. Luo, W. P. Tay and M. Leng, Identifying Infection Sources and Regions in Large Networks, IEEE Transactions on Signal Processing, Vol. 16, No. 11, June 2013, pp. 2850-2865.
[3] O. Trullols-Cruces, M. Fiore and J. M. Barcelo-Ordinas, Worm Epidemics in Vehicular Networks, IEEE Transactions on Mobile Computing, Vol. 14, No. 10, October 2015, pp. 2173-2187.
[4] V. Karyotis and S. Papavassiliou, Macroscopic Malware Propagation Dynamics for Complex Networks With Churn, IEEE Communications Letters, Vol. 19, No. 4, April 2015, pp. 577-580.
[5] A. Oprea et. al., Detection of Early-Stage Enterprise Infection by Mining Large-Scale Log Data, 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks pp. 45-56.
[6] P. Wang, et. al., Markov Model of Malicious Code Propagation, Proc. of 2010 International Conference on Innovative Computing & Communication and 2010 Asia-Pacific Conference on Information Technology and Ocean Engineering, pp. 260-263.
[7] D. Dagon, C.C. Zou, andW. Lee, Modeling botnet propagation using time zones, in Proc. 13th Network and Distributed System Security Symp. (NDSS06), San Diego, CA, Feb. 2006.
[8] Y. Xie, V. Sekar, D. A. Maltz, M. K. Reiter, and H. Zhang, Worm origin identification using random moonwalks, in Proc. IEEE Symp. Security and Privacy, Berkeley/Oakland, CA, May 2005.
[9] Q. Wang, Z. Chen, and C. Chen, Darknet-based inference of Internet worm temporal characteristics, IEEE Trans. Inf. Forensics Security, vol. 6, no. 4, pp. 13821393, Dec. 2011.
[10] Cloud Antivirus, Endpoint, UTM, Encryption, Mobile, DLP, Server, Web, Wireless Security, Network Storage and Next-Gen Firewall Solutions |Sophos Data Protection for Business, Accessed August 2025.
[11] M.S.K. Awan, P. Burnap, and O. Rana, Identifying Cyber Risk Hotspots: A Framework for Measuring Temporal Variance in Computer Network Risk, Computers & Security 57, March 2016, pp. 31-46, Elsevier.