Time-Outs and Counters Against Storms

Erol Gelenbe and Omer H. Abdelrahman
Intelligent Systems and Networks Group
Department of Electrical & Electronics Engineering
Imperial College, London SW72AZ, UK
{e.gelenbe,o.abd06}@imperial.ac.uk

***Abstract*—Mobile Networks are becoming the means of access to much of the Cloud infrastructure of the world, and this massive penetration poses new problems related to security. Mobile network attacks known as "signalling storms" which can be launched by malware or by poorly designed apps, can overloads the network's bandwidth at the cell level, the backbone network's signalling servers, and the Cloud servers. Such storms are based on successive service requests both for bandwidth, and ultimately for Cloud access, that then time-out because the compromised mobile then remains inactive. This paper discusses the role of the time-out itself in affecting or mitigating the storm, and then suggests a novel approach to mitigate signalling storms by the use of a counter that will counts the number of successive signalling transitions that involve the time-out, and disconnects the mobile if this number exceeds a threshold. The threshold is used in addition to the time-out itself which is a first line of defence. A simple model allows us to show the effect of the *choice* of the time-out's duration and how it can be optimised and how it impacts both the congestion in number of attacked mobiles and in the amount of requests they generate both to the network and the Cloud, and allows us to derive the optimum value of the counter's threshold that a mobile operator, or a smart mobile, may use to disconnect the mobile from the network when it appears to be malfunctioning or has been compromised. The optimum counter value substantially reduces both the average number of attacked mobiles and the amount of signalling traffic.**

***Index Terms*—Mobile Networks; QoS; Signalling Storms; Malware; Attack Mitigation; Apps with Malfunctions**

I. Introduction

5G systems are challenged by mobile broadband requirements such as video streaming including 3D and playback, together with machine to machine and vehicular communications. All these will demand a lower signalling overhead and better quality of service (QoS) to short payloads, at much higher traffic volumes and bandwidth requirements.

The constant access to the Cloud in order to offload, away from the mobile devices, the energy and computation critical applications, and the possible use of Clouds to virtualise mobile network control and management, create yet another need for continuous and secure connectivity.

Unfortunately, mobile networks are vulnerable to signalling denial-of-service (DoS) attacks which overload the control plane through small but frequent communications that exploit vulnerabilities in the signalling procedures used, for example, in paging [1], service requests [2] and radio resource control (RRC) [3], [4]. Such attacks can be carried out either by compromising a large number of mobile devices, or from the Internet by targeting a hit list of mobile devices through carefully timed transmissions, and can seriously compromise the connections between a large set of mobiles and the Clouds to which they are connected or are trying to connect, by overloading both the mobiles and the Cloud.

Since, security and uninterrupted connectivity in all mobile applications, will become even more important, not just from the "nuisance" and QoS perspective, but also because it is expected that safety critical applications [5], [6] will transition to Clouds and mobile devices and away from special purpose private networks or the commonly used fixed sensor networks, the whole issue of how network attacks can be detected and mitigated will become even more important. Safety critical applications that will be supported by Clouds and accessed via mobiles will include emergency management, the Smart Metering, Smart Grid Control, public transportation control systems (including railways), and electric vehicle charging networks.

Machine to machine applications will in particular be quite vulnerable to such attacks since they will not have the human-in-the-loop who can turn off a mobile when she sees something strange going on. Thus significant efforts need to be made to better understand the security liabilities and weaknesses of mobile connections to backbone networks and the Cloud and find new approaches to make them resilient and reliable in the face of malicious malware or malfunctioning apps [7].

This paper focuses on attacks known as signalling storms which can result from actual malicious attacks or from malfunctions in apps. We first examine the role of the time-out that typically acts as a first line of defence by limiting the ability of nuisance apps and malware to acquire bandwidth. However we show that though the time-out is useful, its impact is quite limited and better schemes are needed. We then suggest a novel and simple scheme based on counting the number of nuisance transitions between low and high bandwidth states. Such transitions can be used to decide to "detach" mobile devices that appear to be compromised. A simple model allows us to show that setting this count n to an optimum value can substantially reduce the signalling, bandwidth and resulting Cloud overload caused by such attacks.

II. Related Work

Signalling storms are similar to signalling DoS attacks, but they are mainly caused by misbehaving mobile apps that repeatedly establish and tear-down data connections [8] in

order to transfer small amounts of data. Such "chatty" apps trigger repeated signalling to allocate and deallocate radio channels and other resources, and therefore have a negative impact on the control plane of the network [9]. There are a number of recent high profile cases, e.g. in Korea [10] and Japan [11], where large operators suffered major outages due to popular apps that constantly poll the network even when users are inactive. Ad-based mobile monetisation is another culprit, shown to cause erratic spikes in signalling traffic [12]. Many mobile carriers have also reported [13] outages and performance issues caused by non-malicious but poorly designed apps, yet the majority of those affected followed a reactive approach to identify and mitigate the problem.

Unlike flash crowds which last for a short time during special events and occasions such as New Year's Eve, signalling storms are unpredictable and tend to persist until the underlying problem is identified and corrected. Moreover, while mobile subscribers may tolerate degraded QoS during crowded events, the same does not hold with signalling storms, leading to customer churn and revenue loss. This has prompted the mobile industry to promote best practices for developing network-friendly apps [14], [15].

However, guidelines for app developers by vendors and industry groups are not sufficient, since well-designed apps could also trigger a storm, when an unexpected event occurs in the Internet such as loss of connectivity to a popular cloud service. Indeed, an important feature of smartphones and tablets is the "always-on" connectivity which enables users to receive push messages, e.g. to notify of an incoming email or VoIP call. This feature is maintained by having the device send periodic keep-alive messages to a cloud server, typically every few minutes. However, if the cloud service becomes unavailable, then the mobile device will attempt to reconnect more frequently, generating significantly higher signalling load than normal as has recently been reported [16], [17].

A. Storms and Signalling Overload

The Internet carries a lot of unwanted traffic [18], which includes backscatter noise from remote DoS attacks, scanning worms, viruses [19] and spam campaigns. While the risk of such traffic reaching the mobile network and triggering a storm can be effectively eliminated through properly configured middleboxes, a recent review of 180 mobile carriers [20], [21] found that 51% of them allow mobile devices to be probed from the Internet, by either assigning them public IP addresses, allowing IP spoofing, or permitting device-to-device probing within the network.

Large-scale malware infections where mobile users are the target can generate excessive signalling as a by-product of malicious activity. Examples of such malware are premium SMS diallers, spammers, adware and bot-clients, which are among the top encountered threats on smartphones [22]. Indeed, a recent analysis of the traffic profiles of mobile subscribers in China [23] indicated a positive correlation between the frequency of resource-inefficient traffic and malicious activities in the network, including private data upload, billing fraud and TCP SYN flooding. Thus it is expected that signalling storms will continue to pose challenges with the projected growth in mobile traffic [24] and the advent of machine to machine systems for which existing cellular networks are not optimised [25], [26].

In the interaction between a mobile and the Cloud, a mobile app may request to open a session on the Cloud, the Cloud would respond to enable the app to open the session and then the mobile would not actually request a service and disconnect after a time-out. It could repeat this pattern indefinitely creating a nuisance and unproductive workload for the Cloud, as well as for the backbone network, the signalling servers and the mobile itself. This is the type of activity that we will analyse and attempt to mitigate in this paper.

III. The Model

The analysis and discussion in this paper is conducted using very elementary and well established modelling techniques [27], [28] that are widely used in telephony and teletraffic. We represent the set of normal and attacked mobile calls in the system by a state $s(t)$ at time t as:

$$s(t) = (b, B, C, A_1, a_1, \ \ldots\ , A_i, a_i, \ \ldots) \tag{1}$$

where:

- b is the number of mobiles which are just starting their communication in low bandwidth mode,
- B is the number of unattacked mobiles which are in high bandwidth mode,
- C is the number of unattacked mobiles that have started to transfer or receive data or voice in high bandwidth mode,
- A_i is the number of attacked mobiles which are in high bandwidth mode and have undergone a time-out for $i-1$ times,
- a_i is the number of attacked mobiles which have entered low bandwidth mode from high bandwidth mode after i time-outs,

We assume a Poisson arrival process of rate λ of new "calls" or mobile activations, and a call that is first admitted in state b then requests high bandwidth at rate r. Note that r^{-1} can be viewed as the average time it takes a call to make its first high bandwidth request to the network.

With probability $1-\alpha$ such a call will be of normal type and will then enter state B while, while with probability α it will be an attacked call and will request high bandwidth and hence enter state A_1 indicating the first request for bandwidth that is made by a defectively operating app or malware that can contribute to a storm, or an attacked call.

Once a call enters state A_1, since it is an "attacked" or malware based call, it will not start a communication and will time-out after some time of average value τ^{-1}. Note that the time-out is a parameter that is set by the operator, and in practice it is of the order of a few seconds. After entering state a_1, if the mobile device or operator is very "clever", the call may be detected as being anomalous, and will be removed or blocked from the system at rate β_1, where β_1^{-1} is the average time it takes to understand that this call has the potential to contribute to a storm, and to block the call

from further activation. Note that such a facility for blocking malicious calls does not exist today. Furthermore, it is very unlikely that the system is so smart that it can make this decision correctly regarding the call so early in the game, so typically the call will manage to request high bandwidth and then enter state A_2 at rate r_1.

Proceeding in the same manner, in state A_i the anomalous call will again not start a normal communication, so it will eventually time-out after an average time τ_i^{-1} and enter state a_i, and so on. As a consequence, the rates at which calls enter these states is simply:

$$\begin{aligned} \Lambda_{A_1} &= \alpha\Lambda_b, \\ \Lambda_{a_i} &= \Lambda_{A_i}, \\ \Lambda_{A_{i+1}} &= \Lambda_{a_i}\frac{r_i}{r_i+\beta_i}, \\ &= \alpha\Lambda_b\prod_{l=1}^{i} f_l, \text{ where} \\ f_l &= \frac{r_l}{r_l+\beta_l}, \end{aligned} \tag{2}$$

and Λ_b is the rate at which calls enter state b, which will be determined below from a more detailed analysis. Different calls will interfere each other via (a) the access to limited wireless bandwidth, and (b) possible congestion due to signalling and other traffic in the backbone network. However if we neglect these points as a first approximation, calls act independently of each other so that the average number of calls in each of the "attacked" states, that are denoted by a_i and A_i, is the average arrival rate of calls into the state, multiplied by the average time spent by a call in that state, so that we have:

$$\begin{aligned} N_{A_1} &= \frac{\alpha\Lambda_b}{\tau_1}, \\ N_{A_i} &= \frac{\alpha\Lambda_b}{\tau_i}\prod_{l=1}^{i-1} f_l, \ i>1, \\ N_{a_i} &= \frac{\alpha\Lambda_b}{r_i+\beta_i}\prod_{l=1}^{i-1} f_l, \ i>1. \end{aligned} \tag{3}$$

As a consequence, the total average number of attacked calls becomes:

$$\begin{aligned} N_a &= \sum_{i=1}^{\infty}[N_{a_i}+N_{A_i}], \\ &= \alpha\Lambda_b\sum_{i=1}^{\infty}\{[\prod_{l=1}^{i-1} f_l][\frac{1}{\tau_i}+\frac{1}{r_i+\beta_i}]\}. \end{aligned} \tag{4}$$

The rate parameters r_i are actually congestion dependent since a mobile can only access bandwidth if enough bandwidth for a reasonable level of QoS is available, and if interference will not be excessive. Let W denote the bandwidth that is available in a given cell through the effect of one or more base stations. If we call N_i the *average number* of mobiles that are in state $i \in \{b, B, C, a, A\}$, then the bandwidth availability will depend essentially on N_B, N_C, N_{A_i} because for a total amount of bandwidth in the system at a base station level of say W, the total amount of *available bandwidth* may be expressed as some value $W^* = W - w_1(N_B + N_C + \sum_i N_{A_i}) - w_2(N_b + \sum_i N_{a_i})$ where w_1 and w_2 denote the bandwidth allocated per high and low bandwidth request, respectively. Thus in reality the rates r_i will be "slowed down" as W^* becomes smaller since requests will be delayed or will even remain unsatisfied. The matter is of course more complex, because not only the bandwidth allocation itself but the error probabilities in the channel will be affected by the number of mobiles that are actually communicating via the base stations.

Now with regard to normal or un-attacked calls, once a call requests high bandwidth and enters state B, it will start communicating and this will be expressed as a transition rate κ which takes the call into "communication state" C. From C the call's activity may be interrupted, as when a mobile device stops sending or receiving data to/from a web site, or when a voice call has a silent period, in which case the call will return to state B at rate μ. Similarly, the call may end at rate δ, leaving the system.

From B it may either return to C at rate κ signifying that transmission or reception has started once again, or it may time-out at rate τ and return to state b. Once it returns to state b after a time-out, the call can try again to enter state B or state A as a normal or attacked call, since we have to include the fact that a normal call may become an attacked call after acquiring malware during its "normal" communication with a web site or with another mobile. As a consequence, we can calculate the rates at which the calls enter these normal operating states become:

$$\begin{aligned} \Lambda_b &= \lambda + \frac{\tau}{\tau+\kappa}\Lambda_B, \\ \Lambda_B &= (1-\alpha)\Lambda_b + \frac{\mu}{\mu+\delta}\Lambda_C, \\ \Lambda_C &= \frac{\kappa}{\kappa+\tau}\Lambda_B, \end{aligned} \tag{5}$$

which yields:

$$\begin{aligned} \Lambda_B &= \gamma\Lambda_b, \text{ where} \\ \gamma &= \frac{1-\alpha}{1-\frac{\mu\kappa}{(\mu+\delta)(\kappa+\tau)}}, \text{ and} \\ \frac{\tau}{\tau+\kappa}\gamma &= \frac{\tau(1-\alpha)}{\tau+\kappa-\frac{\mu\kappa}{\mu+\delta}}, \\ \Lambda_b &= \frac{\lambda}{1-\frac{\tau}{\tau+\kappa}\gamma}, \text{ so} \\ &= \frac{\lambda}{1-\frac{\tau(1-\alpha)}{\tau+\kappa-\frac{\mu\kappa}{\mu+\delta}}}, \\ \Lambda_B &= \frac{\lambda\gamma}{1-\frac{\tau}{\tau+\kappa}\gamma}, \\ \Lambda_C &= \frac{\kappa\lambda\gamma}{\kappa+\tau(1-\gamma)}. \end{aligned} \tag{6}$$

IV. The Effect of the Time-Out

The expression for N_a in (4) provides us with insight into how the time-out may be used to mitigate attacks. Indeed, combining the expression for N_a with Λ_b given in (6) when

τ_i, r_i and β_i do not depend on i, we have:

$$\begin{aligned} N_a &= \alpha\lambda \frac{r+\beta+\tau}{\beta\tau[1-\frac{\tau(1-\alpha)}{\tau+\kappa-\frac{\mu\kappa}{\mu+\delta}}]}, \\ &= \alpha\lambda \frac{r+\beta+\tau}{\beta\tau[1-\frac{\tau(1-\alpha)}{\tau+\frac{\delta\kappa}{\mu+\delta}}]}, \ or \\ &= \alpha\lambda \frac{(r+\beta+\tau)(\tau+\frac{\delta\kappa}{\mu+\delta})}{\beta\tau(\alpha\tau+\frac{\delta\kappa}{\mu+\delta})}. \end{aligned} \tag{7}$$

so that we may study how N_a varies with τ. In particular, we easily see that:

- As $\tau \to 0$, the effect the time-out is removed since it is infinite, and we have $N_a \to +\infty$ which indicates that the number of attacked mobiles will grow indefinitely because a finite time-out helps to identify and eliminate the attacked mobile devices.
- If $\tau \to +\infty$ the time-out is very fast and $N_a \to \frac{\lambda}{\beta}$. Note that λ which is the rate of incoming calls may be quite high in the thousands of calls per minute, while β^{-1} is the average time it takes to decide that a given mobile has been attacked, and may take minutes. As a result, their product $\lambda\beta^{-1}$ may also be quite high.

Thus it will be better to choose an optimum value of τ between these two extremes, which helps to minimise the total number of attacked mobiles N_a. When we take the derivative of (8) we remain with a second degree equation in τ, the solution of which yields:

$$1/\tau^*_{N_a} = \begin{cases} \frac{\sqrt{(1-\alpha)[\frac{\kappa\delta}{(\mu+\delta)(\beta+r)}-\alpha]}-\alpha}{\frac{\kappa\delta}{\mu+\delta}}, & \text{if } \frac{\kappa\delta}{(\mu+\delta)(\beta+r)} > \frac{\alpha}{1-\alpha}, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

A simple order of magnitude estimate will tell us that $\delta << \mu$ since a complete call will typically be much longer than the time between successive accesses to a web site, or "silent" periods within an interaction from a mobile device can be numerous but short in comparison with the duration of the call as a whole. Similarly, we can assume that $r >> \beta$ since the time it will take to identify and eliminate an attacked call will be much longer than the time needed to request high bandwidth once the call is initiated. Finally, κ may be of the same order of magnitude to r or much smaller, because the transmission times that are represented by κ^{-1} are very short if the device is downloading or uploading bursts of data, but may be much longer (i.e. κ much smaller) if the mobile device is downloading video streams. Thus we can expect that in practice the condition

$$\frac{\kappa\delta}{(\mu+\delta)(\beta+r)} > \frac{\alpha}{1-\alpha},$$

that guarantees the existence of a non-zero minimum value is only satisfied for quite a small value of the attack probability α. This is illustrated in Figure 1, which justifies the use of a small time-out of the order of a few seconds, since it will minimise the value of N_a when α is very small, but keep N_a small also for larger values of α. Figure 2 on the other hand shows the effect of the time-out on N_a for a wider range of parameters and for two different values of β. Again,

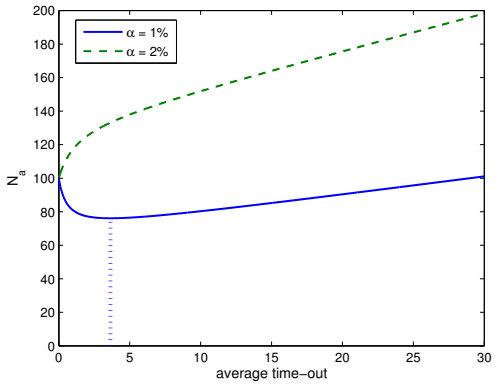

Fig. 1. Number of attacked calls, versus the average time-out τ^{-1} in seconds, for $\alpha = 0.01$ and $\alpha = 0.02$, when $\lambda = 10$, $r = 1$, $\kappa = 1$, $\delta = 1/300$, $\mu = 0.2$ for $\beta = 0.1$.

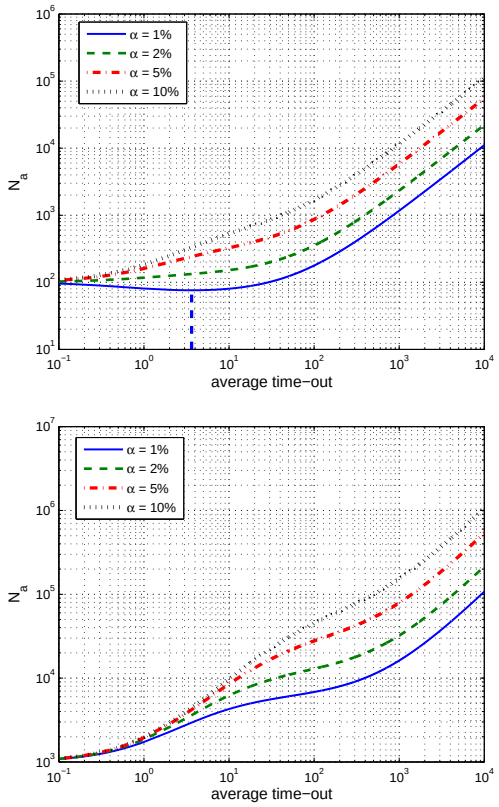

Fig. 2. Number of attacked calls, versus the average time-out τ^{-1} in seconds, for different percentages of attacked devices α, when the various rates per seconds are $\lambda = 10$, $r = 1$, $\delta = 1/300$, $\mu = 0.2$ for (a) $\kappa = 1$, $\beta = 0.1$ (top) and (b) $\kappa = 0.1$, $\beta = 0.01$ (bottom).

we observe a strictly positive value of the optimum time-out for small attack probability α but in other cases, as the average time-out increases, so does the total average number of attacked mobiles.

A. Time-Out and Signalling Load

Denote by Λ the total number of state transitions per unit time, which can be used as a measure of signalling load on

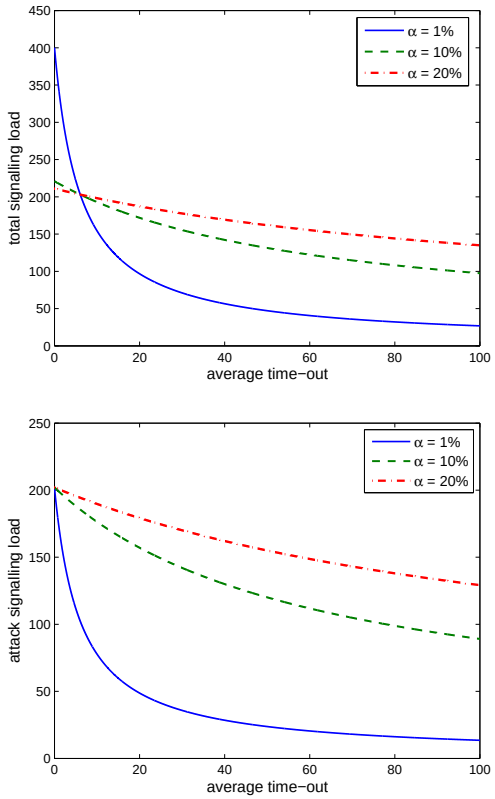

Fig. 3. (a) Total signalling load in requests per second (top), and (b) attack signalling load (bottom) in requests per second versus the average time-out τ^{-1} in seconds for different percentages of attacked devices α, when $\lambda = 1$, $r = 1$, $\kappa = 0.1$, $\delta = 1/300$, $\mu = 0.2$ and $\beta = 0.01$. All rates are normalised to $1/seconds$.

the network, and is given by:

$$\Lambda = \Lambda_b + \Lambda_B + \Lambda_c + \sum_{i=1}^{\infty}[\Lambda_{A_i} + \Lambda_{a_i}]$$
$$= \lambda \frac{\alpha\tau[1 + \frac{2}{\alpha} + \frac{2r}{\beta}] + \frac{\kappa\delta}{\mu+\delta}[3 + 2(1-\alpha)\frac{\mu}{\delta} + \frac{2r\alpha}{\beta}]}{\alpha\tau + \frac{\kappa\delta}{\mu+\delta}}.$$

Notice that Λ becomes *independent* of the time-out rate τ when the condition $\frac{\mu}{\delta} = \frac{1}{\alpha} + \frac{r}{\beta}$ is satisfied; otherwise, the optimum value of the time-out which minimises Λ is:

$$1/\tau^*_\Lambda = \begin{cases} \infty, & \text{if } \frac{\mu}{\delta} < \frac{1}{\alpha} + \frac{r}{\beta} \\ 0, & \text{if } \frac{\mu}{\delta} > \frac{1}{\alpha} + \frac{r}{\beta} \end{cases} \quad (9)$$

In current mobile networks where $\beta \to 0$ we see that longer time-outs reduce the signalling load on the network as one would expect. However, if mobile networks have a scheme for identifying and stopping the activity of mobile devices that have come under attack, such that $\frac{\mu}{\delta} > \frac{1}{\alpha} + \frac{r}{\beta}$, then shorter time-outs will accelerate the blockage of devices that have been compromised, effectively reducing the signalling load on the network.

V. Optimum Counter for Mitigation

Although choosing a relatively small value of the time-out of the order of a few seconds is indeed useful, we see that some additional mechanism needs to be inserted to mitigate the effect of signalling storms. Therefore we suggest that a counter value n be selected so that as long as the number of *successive times* that the mobile uses the time-out is *less than* n, then the mobile remains attached to the network. However as soon as this number reaches n, then the mobile is detached after a time of average value β^{-1}. Thus β^{-1} can be viewed as the decision time plus the physical detachment time that is needed.

Based on this principle, and with reference to our earlier definition of β_i, we have:

$$\beta_i = \begin{cases} 0, & 1 \le i < n, \\ \beta, & i \ge n \end{cases}$$

so that storm mitigation is activated when high bandwidth is requested n *successive* times, each followed by a time-out. Using the previous analysis, the number of average number of attacked calls becomes:

$$N_a = \alpha\Lambda_b[(n - 1 + \frac{r}{\beta})(\frac{1}{\tau} + \frac{1}{r}) + \frac{1}{\tau}] \quad (10)$$

while the resulting signalling rate from the attack is:

$$\Lambda_a = \alpha\Lambda_b + \sum_{i=1}^{\infty}[\Lambda_{a_i} + \Lambda_{A_i}] = \alpha\Lambda_b[2n + 1 + \frac{2r}{\beta}] \quad (11)$$

A large value of n will improve the chances of correctly detecting a misbehaving mobile user, providing mitigation with full confidence to detach the misbehaving mobile from the network. If n is small we may have false positives, requiring analysis of the user's behaviour with other ongoing connections, or checking some data plane attributes such as destination IP addresses or port numbers that may be associated with malicious activities. Thus the higher the n, the faster the decision can be to disconnect the mobile, i.e. β increases with the threshold n, with a slope or derivative with respect to n expressed as $\beta^{'}$.

Indeed with some further simple analysis we can show that the value n^* that minimises *both* N_a and Λ_a, is the value that satisfies:

$$\left(\beta(n^*)\right)^2 \approx r.\beta^{'}(n^*). \quad (12)$$

Figure 4 shows N_a and Λ_a versus n when $\beta(n) = 0.02n$ with $r = 0.5\ secs^{-1}$, and we see that $n^* = 5$ as predicted by (12).

VI. Conclusions

The recent explosive growth in mobile data traffic is marked by an even greater surge in signalling loads due to interworking between the entire mobile ecosystem including devices, apps, network configuration, cloud services and users. As mobile devices and apps increasingly access the Cloud in order to offload computationally intensive or energy-costly activities, and machine to machine applications also exploit the Cloud for storage and decision making, signalling storms can become a significant show stopper both from the Cloud perspective (due to nuisance connections that exploit the Cloud front end) and from the perspective of the response time needs of apps themselves, and excessive congestion in the network. Thus we suggest a novel mitigation approach in which a counter

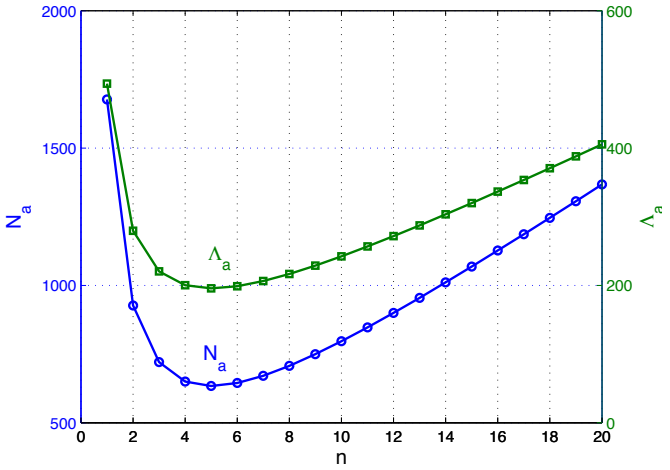

Fig. 4. Number of attacked mobiles (left) and resulting signalling overload (right) versus the number of false transitions that triggers the mitigation mechanism, when $\lambda = 10\ calls/s$, $\tau^{-1} = 5s$, $\alpha = 0.1$, $r^{-1} = 2s$, $\kappa^{-1} = 10s$, $\delta^{-1} = 5$mins, and $\mu^{-1} = 5s$.

is maintained for each mobile device to detect an excess in the number of successive state transitions that time-out, so that misbehaving mobiles can then be detached from the network. This joint detection and protection system can be implemented either at the network signalling server, or directly on the the mobile phone to eliminate the problem at its root. A simple model has allowed us to first examine the role of time-out that typically limits the effect of storms, and then show that the simple counter-based approach substantially reduces the signalling load and nuisance Cloud accesses caused by such attacks. Based on these results, further simulation studies and interaction with standards committees can advance these ideas into a practical scheme that may be used to protect boththe network and the Cloud from such signalling attacks or malfunctions..

REFERENCES

[1] J. Serror, H. Zang, and J. C. Bolot, "Impact of paging channel overloads or attacks on a cellular network," in *Proc. 5th ACM W'shop Wireless Security (WiSe'06)*, LA, CA, Sep 2006, pp. 75–84.

[2] P. Traynor, M. Lin, M. Ongtang, V. Rao, T. Jaeger, P. McDaniel, and T. La Porta, "On cellular botnets: measuring the impact of malicious devices on a cellular network core," in *Proceedings of the 16th ACM conference on Computer and communications security (CCS '09)*. Chicago, Illinois, USA: ACM, 2009, pp. 223–234.

[3] F. Ricciato, A. Coluccia, and A. D'Alconzo, "A review of DoS attack models for 3G cellular networks from a system-design perspective," *Comput. Commun.*, vol. 33, no. 5, pp. 551–558, Mar 2010.

[4] O. H. Abdelrahman and E. Gelenbe, "Signalling storms in 3G mobile networks," in *Proc. IEEE International Conference on Communications (ICC'14)*. Sydney, Australia: IEEE, June 2014, pp. 1023–1028.

[5] A. Filippoupolitis and E. Gelenbe, "A distributed decision support system for building evacuation," in *Proc. 2nd IEEE Conf. Human System Interactions (HSI'09)*, May 2009, pp. 323–330.

[6] E. Gelenbe and F.-J. Wu, "Large scale simulation for human evacuation and rescue," *Computers and Mathematics with Applications*, vol. 64, no. 12, pp. 3869–3880, 2012.

[7] O. H. Abdelrahman, E. Gelenbe, G. Görbil, and B. Oklander, "Mobile network anomaly detection and mitigation: The nemesys approach," in *Information Sciences and Systems 2013*, vol. 264. LNEE Springer, 2013, pp. 429–438.

[8] NSN Smart Labs, "Understanding smartphone behavior in the network," White paper, Jan 2011. [Online]. Available: http://goo.gl/jMtXu

[9] C. Schwartz, T. Hoßfeld, F. Lehrieder, and P. Tran-Gia, "Angry apps: The impact of network timer selection on power consumption, signalling load, and web QoE," *Journal of Computer Networks and Communications*, vol. 2013, no. 176217, 2013.

[10] M. Donegan, "Operators urge action against chatty apps," Light Reading Report, Jun 2011. [Online]. Available: http://goo.gl/vjLf1T

[11] Rethink Wireless, "DoCoMo demands Google's help with signalling storm," Jan 2012. [Online]. Available: http://goo.gl/pQjsAm

[12] S. Corner, "Angry birds + android + ads = network overload," Jun 2011. [Online]. Available: http://goo.gl/2dSf9F

[13] Arbor Networks, "Worldwide infrastructure security report," 2012. [Online]. Available: http://goo.gl/2GZpP

[14] S. Jiantao, "Analyzing the network friendliness of mobile applications," Huawei, Tech. Rep., Jul 2012.

[15] GSMA, "Smarter apps for smarter phones!" Feb 2012. [Online]. Available: http://www.gsma.com/technicalprojects/wp-content/uploads/2012/04/gsmasmarterappsforsmarterphones0112v.0.14.pdf

[16] G. Reddig, "OTT service blackouts trigger signaling overload in mobile networks," Sep 2013. [Online]. Available: http://goo.gl/tJDx9p

[17] Y. Choi, C. hyun Yoon, Y. sik Kim, S. W. Heo, and J. Silvester, "The impact of application signaling traffic on public land mobile networks," *IEEE Commun. Mag.*, vol. 52, no. 1, pp. 166–172, Jan 2014.

[18] F. Ricciato, P. Svoboda, E. Hasenleithner, and W. Fleischer, "On the impact of unwanted traffic onto a 3G network," in *Proc. 2nd Int. W'shop Security, Privacy and Trust in Pervasive and Ubiquitous Computing (SecPerU'06)*, Lyon, France, Jun 2006, pp. 49–56.

[19] E. Gelenbe, "Dealing with software viruses: a biological paradigm," *Information Security Technical Report*, vol. 12, no. 4, pp. 242–250, 2007.

[20] Z. Wang, Z. Qian, Q. Xu, Z. Mao, and M. Zhang, "An untold story of middleboxes in cellular networks," in *Proc. ACM SIGCOMM*, Toronto, Canada, Aug 2011, pp. 374–385.

[21] Z. Qian, Z. Wang, Q. Xu, Z. M. Mao, M. Zhang, and Y.-M. Wang, "You can run, but you can't hide: Exposing network location for targeted DoS attacks in cellular networks," in *Proc. Network and Distributed System Security Symp. (NDSS'12)*, San Diego, CA, Feb 2012, pp. 1–16.

[22] D. Maslennikov, "Mobile malware evolution: Part 6," Kaspersky Lab, Tech. Rep., Feb 2013. [Online]. Available: http://goo.gl/rXJ8J

[23] J. Li, W. Pei, and Z. Cao, "Characterizing high-frequency subscriber sessions in cellular data networks," in *Proc. IFIP Networking Conf.*, Brooklyn, NY, May 2013, pp. 1–9.

[24] Cisco, "Cisco visual networking index: Forecast and methodology, 2013-2018," White Paper, Jun 2014. [Online]. Available: http://goo.gl/xoBrTA

[25] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, and J. Wang, "A first look at cellular machine-to-machine traffic: Large scale measurement and characterization," *SIGMETRICS Perform. Eval. Rev.*, vol. 40, no. 1, pp. 65–76, Jun 2012.

[26] 3GPP, "Machine-type and other mobile data applications communications enhancements (release 12)," 3GPP Mobile Competence Centre c/o ETSI, 650 route des Lucioles 06921 Sophia-Antipolis, FRANCE, TR 23.887, Dec 2013.

[27] E. Gelenbe and R. R. Muntz, "Probabilistic models of computer systems i (exact results)," *Acta Informatica*, vol. 7, no. 1, pp. 35–60, 1976.

[28] E. Gelenbe, "The first decade of g-networks," *European Journal of Operational Research*, vol. 126, no. 2, pp. 231–232, October 2000.