Modelling Interdependency in DDoS Impact Propagation in Smart Grids

Dilara Acarali[a], K. Rajesh Rao[a,b,*], Muttukrishnan Rajarajan[a], Doron Chema[c], Mark Ginzburg[c]

[a]*School of Mathematics, Computer Science & Engineering, City, University of London, UK.*
[b]*Department of Information and Communication Technology, Manipal Institute of Technology, MAHE, Karnataka, INDIA.*
[c]*Technical Team, L7 Defense, BeerSheva, Israel*

Abstract

The traditional power network has now evolved into the smart grid, where cyber technology enables automated control, greater efficiency, and improved stability. However, the integration of IT technologies exposes critical infrastructure to potential cyber-attacks. Furthermore, the interdependent nature of the smart grid's composite networks (IT and OT) means that vulnerability extends across interconnected devices and systems. A DDoS attack is relatively easy to deploy but capable of being highly disruptive, and strategic DDoS attacks against the smart grid can be particularly egregious. In this paper, the *F-C* compromise propagation model is proposed, alongside a behavioural DDoS model, to study the relationships between interdependency, DDoS dynamics, and attack-driven compromise propagation. The model is thoroughly tested and mathematically explored, before being validated against simulations conducted with cyber-security providers L7 Defense.

Keywords: Smart grids, DDoS modelling, DDoS mitigation, Smart grids, Critical infrastructure, Epidemiology.

1. Introduction

Interoperability, and the subsequent characteristic of interdependency, are key features of modern critical infrastructure systems. Interoperability describes the collaborative functionality of two or more systems, most likely involving an exchange of information [1] [2]. Interdependency can then be defined as the reliance of these systems on each other (and the data flows produced) for their own functionality.

In the smart grid, which is defined as a cyber-physical system by NIST [3], the term refers to the connection between information technology (IT) and operational technology (OT). The former, also referred to as 'cyber', enables communication for monitoring, control, and maintenance. The latter enables direct interfacing and management of physical systems and processes. For its own functionality, an IT or OT entity can depend on one or more entities of the same or opposite technology. If the requisite entities fail, normal operations are disrupted in the dependent entities as well. As such, interdependency can make a critical system vulnerable to cascading or propagating failures. It is therefore essential to develop an understanding of the role of interdependency in network security.

With the IT network, the previously-offline and private grid becomes remotely accessible at various points (to varying degrees) for the sake of end-to-end communication. This exposes it to cyber-attacks launched remotely and designed to disrupt or manipulate communication flows. The use of IoT for distributed and machine-to-machine (M2M) data sharing introduces the vulnerabilities of WSNs (Wireless Sensor Networks) into the smart grid and creates a larger attack surface [4]. Furthermore, the smart grid inherits the vulnerabilities and threats typically associated with conventional TCP/IP networks,

*Corresponding author
Email address: rajesh.kavoor@manipal.edu (K. Rajesh Rao)

which may now have unique implications and expanded impact.

One such inherited threat is that of DDoS (Distributed Denial-of-Service) [4] [5] [6] [7]. DDoS is an attack on the availability and functionality of systems and services, and can be categorised as a network-based attack [4]. It involves the generation of malicious packets, sent by many distributed systems to some target. The victim is typically an Internet-facing server providing remote access to systems or services [8]. The nature of the DDoS packets determines the type of attack. For example, a flooding attack may involve packets generated in very large quantities to overwhelm the victim. A crafted attack uses packets designed to consume and hold resources. Attacks can also be classified according to the Internet protocol stack. Application layer attacks can exploit HTTP/HTTPS, whilst transport layer attacks disrupt TCP and UDP. Network interdependency means that service disruptions can have reverberating impact on other processes. Given that system continuity and stability is paramount for electrical power supply networks, attacks on availability must be mitigated to prevent failures and outages.

In order to develop effective smart grid defences, we need to understand how DDoS may lead to system-wide compromise. Specifically, we wish to investigate how a DDoS attack targeting the IT network can cause failures within the IT network, and subsequently also within the dependent OT network. To this end, we propose the novel *F-C* (Functional-Compromised) model, which uses a tiered state-based approach where each tier represents a different subset of the smart grid population. These populations are modelled as a pair of interdependent graphs that are used to define the relationships between the tiers.

We also propose a new DDoS depletion model, which is connected to one of the *F-C* tiers to relate the properties of a DDoS instance to resource degradation and consequent failures. Using the proposed *F-C* model, we aim to examine the scale and scope of compromise across the grid, given the network structure and the DDoS instance.

To summarise, our contributions are as follows:

- A stochastic and tiered state-based *F-C* model of system failure with network interdependency, tracking the spread of compromise driven by a DDoS attack.
- A dynamic process model of DDoS-driven resource depletion on targeted nodes for the attack duration, including incremental recovery.
- Testing and validation of the models mathematically and via simulations of the AMI network provided by industry partners L7 Defense.

This research is a part of Energy Shield [9], a project funded by the European Union's H2020 initiative with the aim of developing toolkits and processes for the improved cyber-defence of smart grids. The project is a collaborative effort between the energy sector, the cybersecurity industry, and academia. Our contribution to Energy Shield is in the focus area of DDoS mitigation. The work presented in this paper is a continuation of the research in [10]. The *F-C* model is an enhanced iteration of the earlier *S-A-C*, a deterministic model based on epidemiological techniques and assuming homogenous mixing and the continuous targeting of the entire IT network population [10]. This was one half of a pair of tools designed to estimate the scale of attack-driven compromise. The *F-C* model seeks to improve on this by considering specific targeting and mixing based on contact probability.

The paper is organised as follows. In Section 2, we present work related to this research, highlighting key pieces and discussing their contributions in context with our own. In Section 3, we provide a definitions and descriptions of the DDoS and *F-C* models. In Section 4, testing exercises and results are detailed, including sensitivity analysis, numerical simulations, and validation against AMI simulations. A discussion of the work is provided in Section 5, considering model behaviours and the implications of our findings. Finally, we summarise and conclude in Section 6.

2. Related Work

Research into smart grid cyber-security can be broadly separated into works that focus on the role played by the underlying grid infrastructure, or on the characteristics and detection of possible attacks. The former tend to centre on the role of network interdependency and cascading failures, which are primarily explored using graph theory and network percolation. These works model the power

and communication networks as a pair of interconnected graphs, and are often based on real life grid designs or incidents.

In their foundational paper (based on an Italian blackout in 2003), Buldyrev et al. [11] analytically modelled cascades using first order percolation phase transition. A single node is initially removed (along with its edges). Nodes losing connectivity as a result are removed in the next iteration, and the process continues as such. At the end, the giant amongst mutually connected clusters represents any remaining functionality. Ruj and Pal [12] used a similar approach to compare the impact of random and targeted attacks, where attack probability increases with node degree. They also considered the remaining giant components to assess the proportion of functionality left after the attack-triggered cascade.

Using graphs based on real networks in the USA, Brummitt et al. [13] studied load shedding between grids. Using the Bak-Tang-Wisenfeld sandpile method, they modelled random load distribution, where nodes shed load if their individually-assigned thresholds are exceeded. The failure count is then estimated using multi-branching, where failure probability for a node is higher for smaller degrees. These papers exemplify the graph-based approach, which is good for approximating real networks and can be explored via simulation models. However, these models may also be complicated to produce and costly to run, depending on the size of networks, whilst being undetailed in terms of impacting factors.

Some researchers have incorporated additional modelling to capture the influence of grid-specific dynamic processes. Rosato et al. [14] combined their graphs (also based on Italian smart grids) with a pair of dynamical models for power and traffic flows, the latter based on one presented in [15]. These were used to assess the quality of service on their respective networks.

Inspired by IEEE 39-Bus and Chinese Guangdong structures, Cai et al. [16] used graphs of data dispatching networks in mesh and double-star topologies. They incorporated a dynamic power flow model to calculate power redistribution caused by failures, and a data exchange model to characterise the data flows. A cascade is triggered if the issued commands are not received before more lines become overloaded.

Poulin and Kane [17] developed a new simulation approach to model interdependency in heterogeneous networks. They created a simulation graph to capture event relationships, which is used to support a dependency network graph. They then produced a counterfactual event graph (CEG) to explore different outcomes for events, and subsequently for the whole simulation.

These works add depth to the basic graph-based approach by considering actual flows rather than just the edges they act through. To create a simpler and more generic tool, we decided to focus on populations, with reference to node connectivity, rather than full-scale graphs. We then added process modelling for incoming traffic to explore the influence of the DDoS attack force.

Epidemiological models are population-based and have been widely applied within cyber-security, specifically where growth is a factor. Past applications include botnet propagation across timezones [18] and worm spread [19] [20]. Many assume mass action, implying homogeneous mixing within the observed population. This is in contrast to the spread of failures captured in graphs, which depend on edges between individuals.

Kenah and Miller [21] combined a stochastic epidemic model with graph theory in an epidemic percolation networks (EPNs). They considered both mass action and contact network-based spread, with a focus on an EPN's degree distribution and component size distributions.

Based on an understanding of a network's degree distribution, we simplify further by summarising contact relationships into mean degree probabilities. Our proposed alternative applies an epidemic-style state machine and splits the target population up into subsets [22], each affected differently. Hence, we use stochastic epidemic modelling like [21] with account of graph structure, but move nodes between states instead of percolating. In this way, compromise spread in any network can be measured at a high level.

This work centres on DDoS as the source of disruption. Existing DDoS research tends to be focused on defensive systems, but some have attempted to capture the dynamics of the attack itself.

Ramanauskaite and Cenys [23] proposed a system of two statistical DoS models for attacks on bandwidth and memory in the presence of a defensive filtering system. They consider the number of channels and their bandwidths, query sizes and arrival rates, traffic bit rates, buffer size and query processing times to define probabilities for bandwidth exhaustion, memory depletion, and blocking

of legitimate traffic (via filtering or being dropped). These are then used to derive a composite attack probability. Similarly, Singh and De [24] developed a statistical DDoS model. Following [23], they focused on bandwidth and the buffer as depletion targets, deriving degradation probabilities for each. These are then combined into a single attack probability. Unlike [23], they also consider inter-arrival rate, positing that total exhaustion is more likely at smaller rates.

Shuaib et al. [25] tested the resiliency of smart meters against attacks, including DoS in the form of ICMP and SYN floods. Their simulation setup featured a smart meter connected via a LAN switch to it's server and a malicious host. They report that the smart meter is easily overwhelmed and functionality severely diminished.

Some works do not focus on DDoS but apply similar methods. Cao et al. [26] used a statistical model to estimate available bandwidth. The model is defined in sections: the packet stream, traffic load, bandwidth, and queues. Each has its own set of parameters. Packets are sequentially placed on a channel to form a stream or are queued within limited-sized buffers. The bandwidth and utilisation are measured, with the latter defined as the proportion of the bandwidth currently in use (i.e. traffic bit rate).

Meanwhile, in their review of methods for measuring smart grid resilience, Das et al. [27] related the Figure of Merit (FOM) approach used by Janic [28] to measure the resilience of rail transport networks to service availability in smart grids. The proposed FOM defines the trajectory of service degradation following an incident, including its minimum value, recovery, and eventual restoration.

These models characterise the traffic flows themselves by identifying the fundamental components they are comprised of, though they may be somewhat generic in application. Using a similar approach, combined with the graph-adjacent dynamical traffic models presented in [14] and [16], we characterise DDoS traffic, and similar to FOM by [28], capture its impact trajectory in the context of the cascades caused.

3. Model Design

The aim of this research is to model failure propagation between interdependent graphs, specifically in the context of smart grids, using state-based systems to capture population dynamics, as an alternative to graph theory and percolation. Our proposed approach is captured in the *F-C* (Functional-Compromised) model. We further narrow the context by specifically considering DDoS-driven failures, where there is a relationship between the scale of the DDoS attack and the rate of failure, and the DDoS model captures the dynamics of an attack against a target. Both models are described in detail in the following subsections.

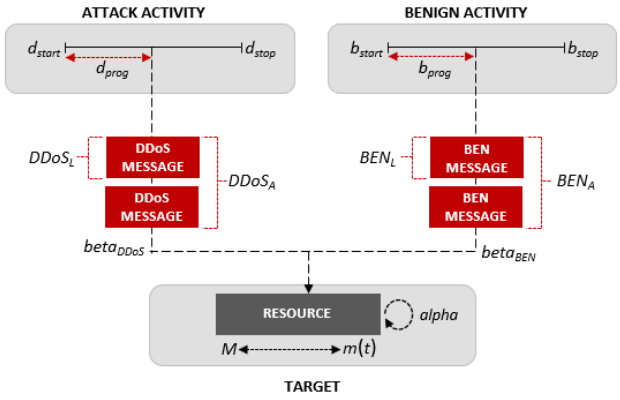


Figure 1: Main components & behaviours of proposed dynamical DDoS model.

3.1. Dynamical DDoS Model

This model captures the impact of incoming traffic on a target which has some finite resource capacity. Formally, given measurements of traffic volume and timings, the model outputs resource availability over time. The nature of the traffic (benign or malicious) depends on the context and the number of input channels is adjustable. The target and the resource are intentionally abstract to allow adaptation for different scenarios. Here, we treat the target as a server with a finite processing capacity [8]. The model, as presented, is deterministic because we wish to find precise outputs for given inputs. However, it can be easily made stochastic by applying sampling of relevant probability distributions for selected parameters.

Figure 1 presents a diagrammatic overview of the DDoS model components and behaviours.

Parameter	Description
M	Target's maximum resource capacity.
α	Target's maximum processing capacity (restoration of resources).
FT	Target's functional threshold.
FC	Target's functional capacity.
d_{start}, b_{start}	Start time of DDoS (d) and benign (b) activity.
d_{stop}, b_{stop}	Stop time of DDoS (d) and benign (b) activity.
d_{dur}, b_{dur}	Duration of DDoS (d) and benign (b) activity.
d_{prog}, b_{prog}	Time since start of DDoS (d) and benign (b) activity.
$DDoS_{B1}(t)$, $BEN_{B1(t)}$	DDoS or benign (BEN) activity start status (Boolean value).
$DDoS_{B2}(t)$, $BEN_{B2(t)}$	DDoS or benign activity stop status (Boolean value).
$DDoS_E$, BEN_E	Turn DDoS or benign activity ON (1) or OFF (0).
$DDoS_L$, BEN_L	DDoS or benign message load (for targeted resource).
$DDoS_A$, BEN_A	DDoS or benign message arrival rate (for target).
β_{DDoS}, β_{BEN}	Overall resource cost of DDoS or benign activity.
$(DDoS)m_D(t)$, $(BEN)m_D(t)$	Resources depleted at time t due to DDoS or benign activity.
$(DDoS)m_R(t)$, $(BEN)m_R(t)$	Resources recovered at time t due to DDoS or benign activity.
$(total)m_D(t)$, $(total)m_R(t)$	Total resource depletions and recoveries at time t.
$m_{delta}(t)$	Net change in available resources at time t.
$m(t)$	Actual available resources at time t.
ω	Estimated traffic activity-based resource depletion rate.
γ	Estimated processing capacity-based resource recovery rate.

Table 1: Summary of parameters in proposed dynamical DDoS model.

3.1.1. Target Characterisation

As mentioned, we do not focus on a particular type of target, only on the units of the resource which is being depleted as a result of the received traffic. The target has a maximum resource capacity of M. The current available resource capacity at a given time is denoted by $m(t)$, which is always some proportion or fraction of the total M. Additionally, $m(t)$ is bounded so that its value is always between 0 and M.

The functional threshold, FT, is the minimum percentage of units needed for the target to remain functional. For example, $FT = 75\%$ means the target can tolerate loss of only 25%. If consumed resources exceed this, it will reach failure state. The functional capacity (FC) is then the proportion of M below which the target fails, and replaces 0 as the value representing complete depletion. If $FT = 0$, then $FC = M$ and the target must reach 0 before it 'fails'. For simplicity, this is considered to be the case for the rest of the model explanations in this section.

3.1.2. Traffic Activity

Within some observation period, the DDoS attack has a start and a stop time, denoted by d_{start} and d_{stop}, which can be used to determine the activity duration d_{dur}. The DDoS period may not be the same as the observation period, so d_{prog} is used to denote the intervals since d_{start}. At each interval of the observation period, the status of the DDoS attack at time t is denoted by $DDoS_{B1}(t)$. The value will be 1 if the attack activity is ongoing at time t, and 0 otherwise. Notification of whether the attack has ended or not is similarly given by $DDoS_{B2}(t)$. The value will be 1 if the attack has started and ended at time t, and 0 otherwise. $DDoS_{B2}(t)$ is relevant when calculating gradual recovery.

The DDoS attack itself can be characterised by its scale. We do not focus on particular type of DDoS, only on its scale. The mean load per DDoS message, in relation to target resources, is given by $DDoS_L$. This is the number of resource units consumed per message. A larger $DDoS_L$ means that each DDoS message is costlier to the

target. The mean DDoS message arrival rate is denoted by $DDoS_A$, which is the number of messages hitting the target in each interval. A larger $DDoS_A$ value results in more messages for the target to deal with in each interval.

Benign traffic activity is defined in the same way as attack activity, so that two traffic streams are established. The model allows for start and stop times to be adjusted, but benign activity is intended to represent the ongoing background activity of a production network.

3.1.3. Activity Impact

In each interval where there is ongoing activity, M is going to be depleted. The number of units depleted by DDoS per interval is denoted by β_{DDoS}, which is the product of $DDoS_L$ and $DDoS_A$. This gives the total unit load of the attack per interval. The units depleted by benign activity is similarly defined as β_{BEN}. The actual units lost per interval, $(DDoS)m_D(t)$, depends on attack start status $DDoS_{B1}(t)$, given by:

$$(DDoS_L\, DDoS_A)\; DDoS_{B1}(t) = \beta_{DDoS}\, DDoS_{B1}(t) \tag{1}$$

This determines what happens once the attack has begun. At t=0, $m(t)$ will be $M - (DDoS)m_D(0)$, and at subsequent times, $m(t)$ will be $m(t\text{-}1) - (DDoS)m_D(t)$. Meanwhile, $DDoS_{B2}(t)$ can be used to determine what happens after the attack has ended. When β is assumed to be constant, $(DDoS)m_D(t)$ can be calculated as:

$$\begin{aligned} DDoS_E\,[((\beta_{DDoS}\, d_{prog})\; DDoS_{B1}(t)) + \\ ((\beta_{DDoS}\, d_{dur})\; DDoS_{B2}(t))] \end{aligned} \tag{2}$$

Here, $DDoS_E$ is an optional parameter with a value of 1 or 0. This simply allows the model user to turn activity types ON or OFF. The same approach is used to find the actual units lost via benign activity per interval, $(BEN)m_D(t)$. Total depletion $(total)m_D(t)$ is then the sum of (DDoS)$m_D(t)$ and $(BEN)m_D(t)$.

In each interval, some units are recovered. This is denoted as α, which is predetermined based on the specific target under consideration. The target has a fixed number of units it can process per time, regardless of the traffic it receives. Thus, the assumption is that an ongoing attack does not diminish this basic processing capacity. Rather, the attack is successful in causing impact when this is overcome. The actual units regained per interval, $(DDoS)m_R(t)$, depends on $DDoS_{B1}(t)$ and $DDoS_{B2}(t)$, and is calculated as:

$$[(\alpha\; d_{prog})\; DDoS_{B1}(t)] + [(\alpha\; d_{prog})\; DDoS_{B2}(t)] \tag{3}$$

Again, recovery for benign activity is calculated in the same way. Total recovery $(total)m_R(t)$ is then the sum of $(DDoS)m_R(t)$ and $(BEN)m_R(t)$. The net change in available units is $m_{delta}(t)$, calculated as $-(total)m_D(t) + (total)m_R(t)$. Then, the target's available resources at time t, $m(t)$, is given by $M + m_{delta}(t)$. Both β and α are assumed to be constant throughout. If, say, the attack is supposed to grow over time, $DDoS_A$ should be measured at each interval and β_{DDoS} recalculated each time. When the attack ceases, depletion activity also stops, i.e. $(DDoS)m_D(t)$ becomes 0. Meanwhile, recovery continues, gradually restoring the target's available resource units. Therefore, $m(t)$ becomes $m(t_{stop}) + (total)m_R(t)$.

3.1.4. Rate Estimations

The previously defined β_{DDoS} and β_{BEN} are the depletion rate of units through attack and benign activity, respectively. The target depletion rate ω, is the rate at which the target itself fails as a result. First, we consider the case where failure state requires resources to reach 0 (FC = 1). When both are constant, time to failure state will be M divided by the sum of the β values, denoted by t_ω. This gives the interval count (or period) needed to go from M to 0. Then, the inverse of this period gives the rate of reaching failure state for the target. Hence, ω denotes how many targets (or how much of a target) is lost per time, calculated as:

$$1/\frac{M}{DDoS_E \times (\beta_{DDoS} + \beta_{BEN})} = \frac{1}{t_\omega} \tag{4}$$

If the functional threshold FT is defined, the target may fail at a value greater than 0. Therefore, failure state is now when the target's resources drop below FC. The target is therefore allowed to lose $M - FC$ resource units before becoming non-functional. Hence, t_ω is amended and ω is calculated as:

$$1/\frac{M - FC}{DDoS_E + \beta_{DDoS} + \beta_{BEN}} = \frac{1}{t_\omega} \tag{5}$$

When FC is greater than 0, ω with FC will be greater than without for the same β because it will take less time to reach the FC value than to reach 0.

The previously defined α is the recovery rate of units. The target recovery rate γ is the rate at which the target itself recovers from the attack. The interval count or period needed to go from 0 to FC is t_γ. Given that above FC is where the target regains functionality, γ will be:

$$1/\frac{FC}{\alpha} = \frac{1}{\gamma} \tag{6}$$

When FC is greater than 0, γ with FC will be larger than without for the same α because it will take less time to reach M from FC than from 0.

The parameters of the dynamical DDoS model are summarised in Table 1.

3.2. Functional-Compromised (F-C) Model

The model captures the impact of a DDoS attack (or similar kind of disruptive event) on a pair of interconnected networks, conceptually realised here as the IT and OT networks of the smart grid. Formally, given properties of the network graphs and the attack, the model outputs the fractions of functional and compromised populations over time. The DDoS attack targets one or a few Internet-facing IT nodes, and disruption to these nodes has an impact on their dependents. Thus, this impact propagates throughout the targeted and dependent networks. The *F-C* model is based on epidemic modelling approaches, and serves as a development on the *S-A-C* model presented in [10].

3.2.1. Network Population

As with standard graph-based approaches, the smart grid is abstracted into a conceptual pair of interconnected graphs. Unlike those approaches, rather than using the graphs directly, their sizes and connections will act as inputs to the model. The IT graph is denoted as $G_X = \{V_X, E_X\}$, where V_X is the set of IT nodes and E_X is the set of intra IT-network edges. The OT graph is denoted as $G_Y = \{V_Y, E_Y\}$, with V_Y representing OT nodes and E_Y representing intra-OT network edges. A third set of edges, E_Z, represents inter-network edges connecting the IT and OT graphs.

Edges are directed and embody a dependency relationship between a pair of nodes. We refer to the source node as the dependent and the sink node as the requisite. In other words, the direction of the edge maps a dependent node to its requisite node. For simplicity, only OT-to-IT dependencies are considered for inter-network edges. The probability λ_{XX} captures the likelihood of a dependent contact between two X (IT) nodes. It is calculated as the number of intra-X edges over the maximum possible edges in G_X as:

$$\frac{E_X}{|V_X|(|V_X|-1)} \tag{7}$$

Similarly, λ_{YY} is the probability of a dependent contact between two Y (OT) nodes, and is calculated in the same way for graph G_Y. For edges between X and Y, λ_{XY} is defined as the probability of dependent contact, and is calculated as follows:

$$\frac{E_Z}{|V_X|(|V_Y|)} \tag{8}$$

The node sets V_X and V_Y determine the population sizes for the model, such that the total population N is $V_X + V_Y$. A subset of the IT graph (X) is defined to hold the IT nodes which are targeted by the DDoS attack. This is denoted by T. This creates 3 sub-populations of N, arranged in three tiers as T, X, and Y. Two possible states for the nodes are defined as functional (F) and compromised (C) and applied across the tiers. Now, attack targets are denoted as FT and CT, IT nodes as F_X and C_X, and OT nodes as F_Y and C_Y. This means $N = F_T + C_T + F_X + C_X + F_Y + C_Y$. The model is presented in the form of a state-flow diagram in Figure 2.

3.2.2. Mathematical System

Epidemic models, and those based on them, consider changes in populations over time, where each population represents some state inhabited by individuals. Each state has a set of characteristics that define the individual's status. An individual moves between states at some rate, and the trajectories of populations are used to predict growth/decay and to define preventative or remediation strategies. This modelling approach is adapted to the *F-C* model.

The follow equations deterministically define the proposed model as a system of differential equations:

$$\frac{F_T}{dt} = -(\omega_A F_T) - (\omega_D (\lambda_{XX} F_T C_T)) + (\gamma C_T) \quad (9)$$

$$\frac{C_T}{dt} = +(\omega_A F_T) + (\omega_D (\lambda_{XX} F_T C_T)) - (\gamma C_T) \quad (10)$$

$$\frac{F_X}{dt} = -(\omega_D (\lambda_{XX} F_X C_T)) - (\omega_D (\lambda_{XX} F_X C_X)) + (\gamma C_X) \quad (11)$$

$$\frac{C_X}{dt} = +(\omega_D (\lambda_{XX} F_X C_T)) + (\omega_D (\lambda_{XX} F_X C_X)) - (\gamma C_X) \quad (12)$$

$$\frac{F_Y}{dt} = -(\omega_D (\lambda_{YX} F_Y C_X)) - (\omega_D (\lambda_{YY} F_Y C_Y)) + (\gamma C_Y) \quad (13)$$

$$\frac{C_Y}{dt} = +(\omega_D (\lambda_{YX} F_Y C_X)) + (\omega_D (\lambda_{YY} F_Y C_Y)) - (\gamma C_Y) \quad (14)$$

where:

- ω_A is the rate at which nodes become non-functional due to DDoS.
- ω_D is the rate at which nodes become non-functional due to dependencies.
- γ is the rate, at which nodes return to functional state by recovering resources.
- λ_{XX} is the degree probability of an IT node having a dependent edge to another IT node.
- λ_{YX} is the degree probability of an OT node having a dependent edge to an IT node.
- λ_{YY} is the degree probability of an OT node having a dependent edge to another OT node.

In this work, we focus on the stochastic implementation of the these equations, as a system of probabilities as follows:

$$P_{T1} = \omega_A F_T \qquad \{F_T - 1, C_T + 1\} \quad (15)$$

$$P_{T2} = \omega_D (\lambda_{XX} F_T C_T) \qquad \{F_T - 1, C_T + 1\} \quad (16)$$

$$P_{T3} = \gamma C_T \qquad \{F_T + 1, C_T - 1\} \quad (17)$$

$$P_{X1} = \omega_D (\lambda_{XX} F_X C_T) \qquad \{F_X - 1, C_X + 1\} \quad (18)$$

$$P_{X2} = \omega_D (\lambda_{XX} F_X C_X) \qquad \{F_X - 1, C_X + 1\} \quad (19)$$

$$P_{X3} = \gamma C_X \qquad \{F_X + 1, C_X - 1\} \quad (20)$$

$$P_{Y1} = \omega_D (\lambda_{YX} F_Y C_X) \qquad \{F_Y - 1, C_Y + 1\} \quad (21)$$

$$P_{Y2} = \omega_D (\lambda_{YY} F_Y C_Y) \qquad \{F_Y - 1, C_Y + 1\} \quad (22)$$

$$P_{Y3} = \gamma C_Y \qquad \{F_Y + 1, C_Y - 1\} \quad (23)$$

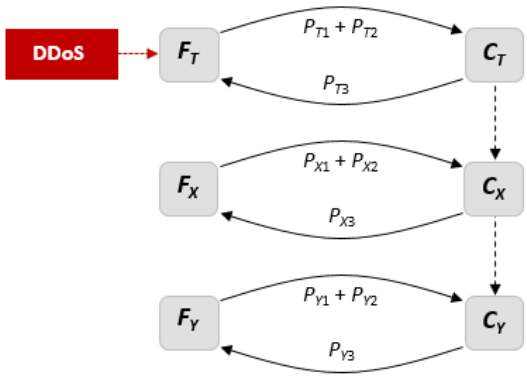


Figure 2: State-flow diagram of the proposed *F*-*C* model.

For both versions, the overall state of N (or the system) at time t is presented as $S = \{F_T(t), C_T(t), F_X(t), C_X(t), F_Y(t), C_Y(t)\}$. Summary descriptions of the probabilities are provided in Table 2.

3.2.3. Transitions

The attack is absorbed by the T tier, with the impact of the attack-based depletion captured by ω_A. This is applied only to the F_T population. Nodes in F_T become compromised by the attack with probability P_{T1} and move into C_T. Meanwhile, the impact of failing requisite nodes is captured in the dependency-based depletion rate ω_D. This is applied to the proportion of $F_T \times C_T$ contacts constituting a dependency relationship, provided by λ_{XX}. Here, nodes become compromised with probability P_{T2} and again move into C_T.

Events happening in the T tier have a consequential impact in the wider IT network (i.e. the X tier) for which some requisite nodes are now in C_T. This means F_X will suffer compromises due to these losses, and nodes will transition into C_X. Hence, F_X loses nodes with the probability P_{X1} and P_{X2} through T-to-X and X-to-X connections, respectively.

Similarly, a similar process unfolds at the Y tier, where events happening in the X tier have an impact. F_Y will suffer compromises due to the loss of requisite nodes, currently sitting in C_X and C_Y. Hence, F_Y loses nodes with probability P_{Y1} and P_{Y2} through X-to-Y and Y-to-Y connections. At each tier, nodes also recover functionality at

Parameter	Description
P_{T1}	T compromise where an F_T node loses functionality via attack.
P_{T2}	T compromise where an F_T node loses functionality via C_T nodes.
P_{T3}	T recovery where a C_T node regains functionality.
P_{X1}	X compromise where an F_X node loses functionality via C_T nodes.
P_{X2}	X compromise where an F_X node loses functionality via C_X nodes.
P_{X3}	X recovery where a C_X node regains functionality.
P_{Y1}	Y compromise where an F_Y node loses functionality via C_X nodes.
P_{Y2}	Y compromise where F_Y node loses functionality via C_Y nodes.
P_{Y3}	Y recovery where a C_Y node regains functionality.

Table 2: Probabilities defined in proposed stochastic F-C model.

the rate γ, and with the probabilities P_{T3}, P_{X3}, and P_{Y3}. Attacked nodes will recover when they are able to free up and regain depleted resources. Conceptually, dependent nodes will recover when their requisite nodes regain functionality.

3.2.4. Parameters

The attack-based depletion rate ω_A is derived from the properties of the DDoS attack, as described in Section 3.1.4. The dependency-based depletion rate ω_D is chosen by the user of the model to allow for network-specific factors to be considered. The deriving of a mathematical definition for ω_D is part of our future planned work.

The recovery rate γ is derived from the properties of the targeted node/s, also described within Section 3.1.4. C_T node restoration depends on how quickly a target is able to recover resources, whereas C_X and C_Y node restoration rate depends on that of C_T nodes. We assume a negligible delay between C_Y and C_X restoration, so that they recover at the same rate as C_T. Hence, the same γ is applied to depletions of both types.

Degree probabilities (λ_{XX}, λ_{YY}, and λ_{YX}) denote the likelihood of contact events being dependency-based. The standard approach in population-based models is to assume mass-action, which means that there is an equal chance of contact between each pair of nodes [29] [21]. However, the model is to be aligned with the properties of the underlying networks. This means that not every contact is actually representative of a dependency relationship. Therefore, the relevant degree probability is applied to the contact rate to scale it appropriately. In this way, we aim to achieve a middle-ground between mass action and network-based propagation [21].

3.2.5. Behaviour Pattern

The analysis of the F-C model's behaviour patterns during a DDoS attack and subsequent recovery is elaborated on in this section. Here, we study the probability of DDoS attack and recovery rates at a given time t to determine the status of the F-C model and also to determine how the recovery pattern gains over the attack pattern during the attack.

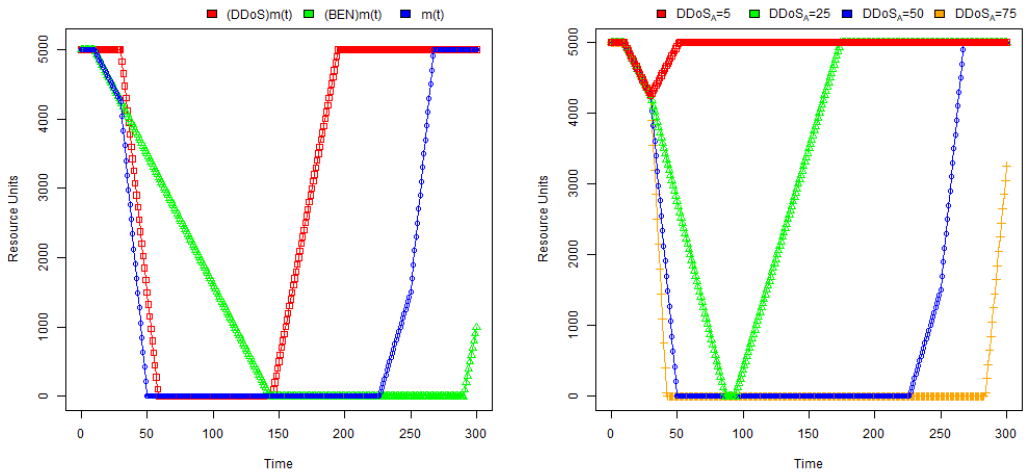
Let BP be the observed F-C model behaviour pattern at time t, so that $o(t) = BP$. Also, let $BP = a + r$, where a corresponds to the attack pattern, and r corresponds to the recovery pattern. Then, the probability of the attack rate is $Pr[attack\ pattern] = \frac{a}{BP}$. Similarly, the probability of the recovery rate is $Pr[recovery\ pattern] = \frac{r}{BP}$. Therefore, at any given point in time t, the model is said to be healthy based on its behaviour pattern if the recovery rate is much greater than attack rate, i.e. $Pr[recovery\ pattern] = \frac{r}{BP} >> Pr[attack\ pattern] = \frac{a}{BP}$.

Furthermore, at time t_i, let r_i be the recovery pattern and a_i be the attack pattern. During the healthy status of the F-C model, it may be checked that at any point of time t_i, $r_i + a_i = BP$, where $r_i \leq BP$ and $a_i \geq 0$ for all i's. The sequence of the recovery pattern $\{r_i\}_{i=0,1,2,\cdots}$ is a monotone increasing sequence. Clearly,

$$\lim_{i\to\infty} \frac{r_i}{BP} = \frac{BP}{BP} = 1 \tag{24}$$

Similarly, the sequence of the attack pattern $\{a_i\}_{i=0,1,2,\cdots}$ is a monotone decreasing sequence. Clearly,

$$\lim_{i\to\infty} \frac{a_i}{BP} = \frac{0}{BP} = 0 \tag{25}$$



(a) Trajectories of DDoS, benign, and cumulative impact.

(b) Impact of changing β_{DDoS} via $DDoS_A$.

Figure 3: Outputs from numerical simulations on dynamical DDoS model.

Therefore, the recovery pattern gains over the attack pattern with time during the DDoS attacks.

4. Testing

The model formulations described in Section 3 were informed by discussions within the Energy Shield project consortium, as well as by our direct collaboration with L7 Defense [30]. The models were tested for functionality, sensitivity, and validity through numerical simulation and using the L7 Defense's AMI simulation environment. In this way, we were able to tune, refine, and expand upon different aspects of the models, and to test them within realistic smart grid scenarios.

4.1. Numerical Simulations

To explore the models in terms of the inputs/outputs and the relationships between parameters, numerical simulations were conducted. These also served as uni-variate sensitivity tests [31]. By establishing a baseline, and then varying values of a single parameter at a time, we were able to assess the models' functionality and identify any indirect relationships.

4.1.1. Setup

To test the dynamical DDoS model, the observation period was set to 300 seconds (5 minutes). Benign activity, representing general background traffic to the target, begins at 10 seconds and stops at 250. This is to allow us to observe any acceleration in recovery when there is no activity. Meanwhile, DDoS activity begins at 30 seconds and stops at 90 seconds, replicating a short burst-style attack. For easy comparison, $DDoS_L$ and BEN_L were both set to 5.5 units per message, assuming that DDoS messages mimic the size of normal messages [8]. For arrival rates, BEN_A was set to 25/s, whilst $DDoS_A$ was more intense at 50/s. On the target, α was set to 100, and maximum resource capacity is set to 5,000 units.

Using this baseline, we performed uni-variate analysis on α, $DDoS_L$, BEN_L, $DDoS_A$, BEN_A, FT, and activity durations. In each case, we observed the trajectory of $m(t)$ for the two activity types individually and collectively. Whilst we performed multiple runs in each case, we did not use Monte Carlo simulation given that the model is deterministic.

To test the F-C model, the observation period was also set to 300 seconds. It is assumed that the attack is ongoing and continues to degrade targets throughout this period. For the baseline, both N_X and N_Y were initialised at

500 nodes each, where $F_T = 10$, $F_X = 490$, $F_Y = 500$. The edge counts (which determine the dependency probabilities λ_{XX}, λ_{YY}, and λ_{YX}) were $E_X = E_Y = E_Z = 500$. Note that a dependency probability of 1 would mean that a dependency exists between every inter- and intra-network node pair. Meanwhile, for simplicity, ω_A and ω_D are both set to 0.5, and γ is set to the slower rate of 0.1.

Using this baseline, we performed uni-variate analyses first (for base parameters) on ω_A, ω_D, and γ, and second (for network parameters) on E_X, E_Y, and E_Z. In each case, we observed the trajectories of the different sub-populations over time. Given that the stochasticity of the process, Monte Carlo simulation and averaging was used.

4.1.2. DDoS Model Results

For the relationship between activity types, when both have the same scale, the one starting earlier can mask the other with its depletion. When depletion events caused by either activity overlap (i.e. occur in the same intervals), the cumulative impact is felt on the target. This is depicted in Figure 3 (a).

As β_{DDoS} is the product of $DDoS_L$ and $DDoS_A$ (and the equivalent is applicable for BEN), increasing either causes a steeper (and earlier) downward trajectory for $m(t)$. However, this only becomes apparent if $m(t)$ has not already been significantly reduced by earlier activity. At very large values for β, where it overcomes α, $m(t)$ drops to minimum value almost instantly. Both parameters also cause increases in ω, as they directly contribute to it. As γ is a function of the target and not the traffic, the upward slope for $m(t)$ recovery remains consistent regardless of traffic scale. Figure 3 (b) demonstrates the impact of changing β via $DDoS_A$, where attack message size and benign traffic scale remains constant.

The recovery trajectory is however influenced by activity duration. Longer d_{dur} or b_{dur} means that a longer recovery period is needed to approach M. Furthermore, the longer the duration, the more likely it is that $m(t)$ reaches 0 (or FC). The upward recovery slope is also pushed later in time, so that services are impacted for longer.

As expected, increasing α leads to a flatter downward trajectory. A larger α causes increases in $(DDoS)m_R(t)$ and $(BEN)m_R(t)$ so that more of M is recovered, and at very large values of α, $m(t) = M$ and remains constant. This is because the target is able to process and recover incoming messages before there is any significant impact on resources. Meanwhile, γ increases with α, as the latter feeds directly into the former.

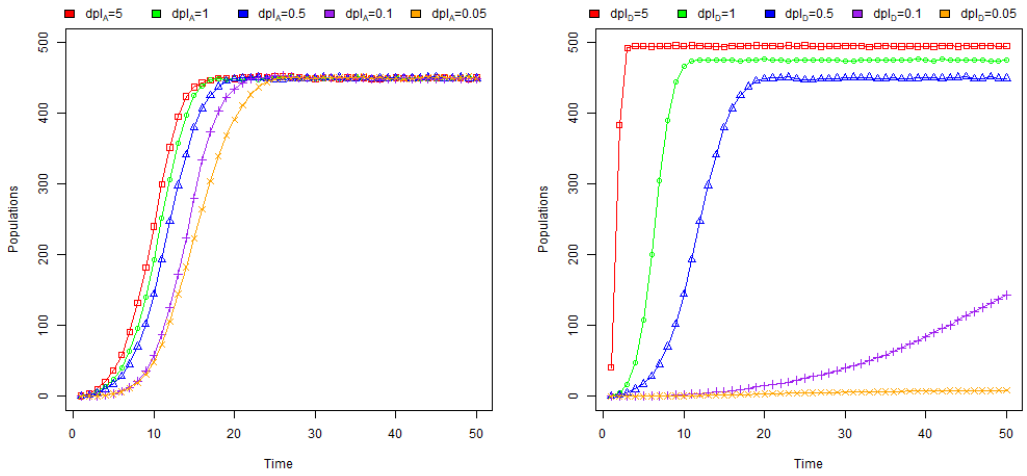
When $FT = 0$, FC also equals 0 (i.e. when $m(t)$ drops below 0, the target will fail). Then, $M - FC = M$, meaning the target can deplete all the way to 0. To return to functionality, $m(t)$ must also reach 0. Hence, the time between the failure point and the recovery point is nil. In this case, γ rate becomes infinite so that the target recovers instantly. Meanwhile, when $FT = 100$, $FC = M$, meaning that the target can tolerate 0% loss. Then, $M - FC = 0$. The time to reach FC will be nil, so that ω rate becomes infinite. Then, when traffic is received, the target fails instantly. To return to functionality, $m(t)$ must increase from 0 to FC, which is equal to M. Increasing FT therefore causes an increase in ω and a decrease in γ.

4.1.3. F-C Model Results

Increasing ω_A generally causes a larger number of compromises to happen sooner, resulting in a steeper decline in the F populations . Across all test cases, C_Y was observed to be greater than C_X. This is likely caused by the fact that the Y-tier represents the whole of network Y, whilst the X-tier represents network X minus the targeted nodes, so when $N_X = N_Y$, $(F_Y + C_Y) > (F_X + C_X)$. Furthermore, the Y-tier has a greater number of dependencies via E_Y (intra-Y connections) and E_Z (inter X-Y connections).

Increasing ω_D similarly causes more compromise events in a shorter amount of time, but with significantly greater impact due to ω_D being applied across all tiers (whereas ω_A is only applied in the T-tier). At very high values for ω_D, decline in F populations happens almost instantly, whilst very low values almost completely diminish any decline. This highlights the significant role of interdependency in compromise propagation. The contrasting impacts of the two ω rates is demonstrated in Figure 4 for C_Y populations (i.e. the furthest that compromises may reach).

As expected, compromise events are also minimised as the value of γ is increased. As this value is reduced, a sharper decline is observed in the F populations, a behaviour that becomes more pronounced as γ drops below both ω_A and ω_D. When $\gamma = \omega_A = \omega_D$, both F_X and F_Y drop by approximately half of their initial populations and then remain constant. An explanation for this is that initial decline is driven by the two ω rates plus a large F population, which overcome the γ rate. Once the available F



(a) Impact of changing ω_A on the C_Y population.

(b) Impact of changing ω_D on the C_Y population.

Figure 4: Outputs from numerical simulations on *F-C* propagation model.

population is reduced, this force is lessened so that the γ rate can sustain the populations at those levels. This effect generally appears stronger in the stochastic *F-C* model compared to the deterministic version.

When E_X is 0 (i.e. there are no intra-X dependencies), there is no activity beyond the T-tier because there is no means for compromises to propagate. As E_X is increased, so do the number of compromises within the X network, and subsequently, in the Y network too. If E_X is larger than E_Y and E_Z, the final size of C_X can surpass that of C_Y. When E_Y is 0 (i.e. there are no intra-Y dependencies), Y network compromises still take place because of dependency links with the X network. Put differently, when E_Y = 0, and E_X = E_Z, F_X and F_Y nodes basically have the same level of interaction with C_X.

When E_Y is increased, compromise possibility within Y also increases, and C_Y subsequently grows larger. On the other hand, when E_Z = 0 (i.e. there are no dependencies between the X and Y tiers), no activity is observed in Y. When E_Z is increased, the number of compromises grows at a greater rate than when E_Y is increased. This is because Y is absorbing the impact of C_X, and then amplifying it internally via E_Y connections. Hence, E_Y appears to be the main driver for compromises in the Y network. This highlights the significant role of internal network dependencies, in addition to inter-network dependencies.

4.1.4. Cross-Model Influence

The dynamical DDoS model is designed to output an estimated resource consumption trajectory for some given attack and target, as well as estimated depletion and recovery rates based on those inputs. To explore possible interactions between the models, we combined them so that attack and target parameters could be used to produce estimated values for ω_A and γ in the *F-C* model. The previously-defined baselines were used for the remaining parameters.

The two models use the same observation period, so that the impact of the ongoing attack can be observed at the population level and on a single target node, but do not currently run simultaneously. Full integration of the two models is part of the planned future work for the Energy Shield project (discussed more in Section 5).

As expected, increases in α were directly proportional to increases in γ, meaning that at higher values, fewer compromises are observed. As previously described, ω_A is infinite when FT = 100, and γ is infinite when FT = 0. Within this range, ω_A grows and γ drops as FT is increased. This causes more compromise events to happen sooner in time. Meanwhile, increasing $DDoS_L$, $DDoS_A$,

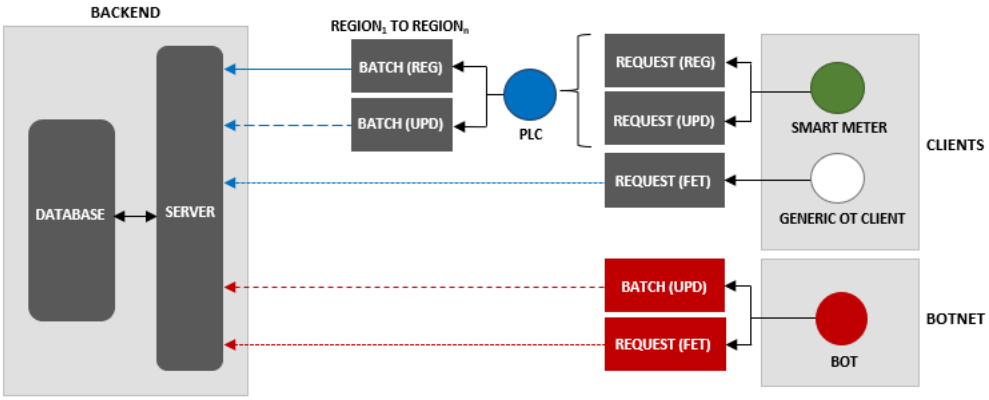


Figure 5: Diagram of the L7D AMI simulation environment structure.

BEN_L, or BEN_A results in a larger value for ω_A too.

From the perspective of the *F-C* model, attack and benign traffic are subsumed within ω_A instead of being considered separately as done by the dynamical DDoS model. The assumption is that normal activity on its own should not lead to node failures. When compared to the corresponding DDoS model outputs, faster declines in the *F* populations of the *F-C* model align with faster depletion of available resource units.

4.2. AMI Simulations

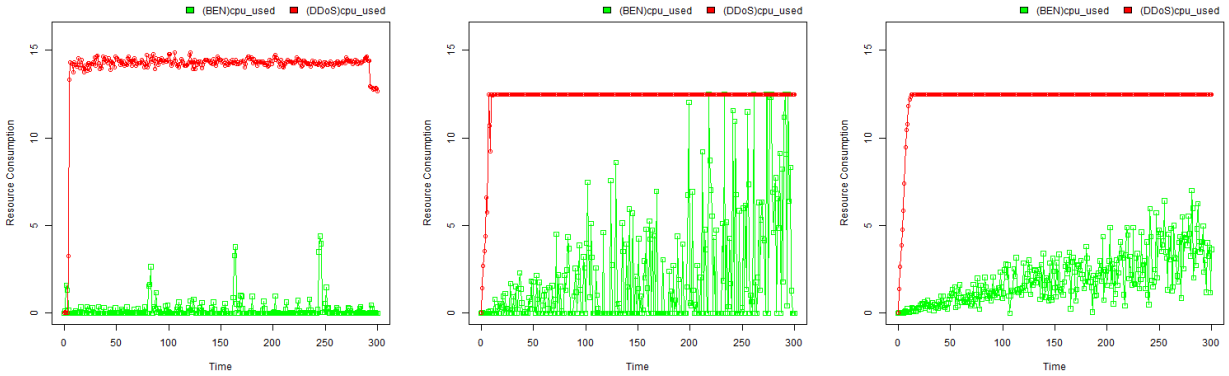
L7 Defense [30] is a software and cyber-defence company who provide AI tools for the protection of control centres and critical infrastructure. They are specialists in defence against large-scale DDoS attacks, and a fellow member of the Energy Shield project consortium. To validate the proposed models, we collaborated with L7 Defense to adapt and use their AMI simulation environment. Based on their professional insights, a number of test cases were defined and the results were then used to assess the predictive capabilities of the dynamical DDoS and *F-C* models.

The Python-based simulation environment is designed to mimic an AMI network. This includes a client layer, an aggregation layer, and a backend layer, as well as a botnet. The backend consists of a server, which receives all incoming requests and processes them, plus an attached SQL database. There are two types of client: smart meters and generic OT endpoints. New smart meters join the network over time, each generating a registration-update request pair. The former contains the meter's identity information, whilst the latter contains its initial energy reading. Both registration and update requests are sent to the aggregator layer where a PLC bundles them into batch requests, which are then forwarded to the backend at given intervals. If successful, a meter's registration details are added to the database. Registered meters continue to send periodic updates on their energy readings over time. Meanwhile, generic OT endpoints in the client layer generate intermittent fetch requests - sent directly to the backend - to retrieve recent consumption data per region. Figure 5 gives a depiction of the environment.

The emulated botnet consists of a churning population of bots so that the attacking IPs change over time. To explore the non-trivial threat of an application layer DDoS attack [8], active bots are given the ability to launch update or fetch floods against the backend, mimicking PLCs, meters, or OT endpoints. In order to test the role of interdependency, the fetch attack scenario (FET-SER) was used in this work. The server itself is directly targeted, allowing us to observe the impact of increased server load on the PLCs and clients. We also used a neutral scenario (NO-ATT), without any botnet activity, as a baseline.

4.2.1. Setup

The AMI setup in this study consists of three PLCs serving a contiguous range of regions, with 30 regions in total and up to 30,000 meters joining the network over



(a) AMI simulation environment.

(b) Dynamical-DDoS model (single run).

(c) Dynamical-DDoS model (mean).

Figure 6: Comparison of resource consumption results generated by the simulator and predicted by the dynamical-DDoS model (where consumption is estimated per available CPU.)

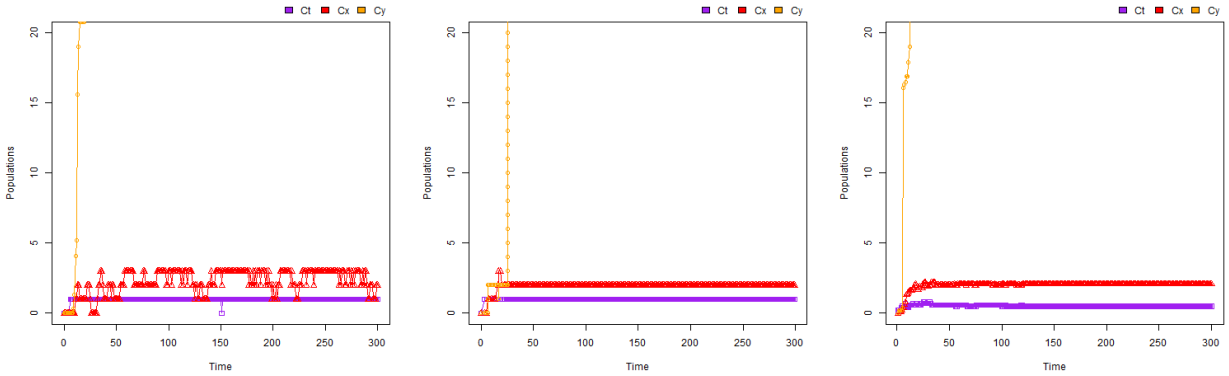
50 minutes. Updates are generated by smart meters every 300 seconds and fetch requests are generated by OT endpoints every 120 seconds. PLCs forward aggregated registrations and updates every 10 and 80 seconds, respectively. The maximum bot count is 20,000, with 1,000 bots active at the start time. Logs are collected on the client, aggregation, and botnet layers for requests generated, received, blocked, and timed-out. Logs are collected on the server for request arrivals and CPU usage.

To explore benign traffic, the simulator was run multiple times in NO-ATT mode, and the mean interval values were calculated. From this, the mean benign message count and mean CPU usage per interval were used to estimate BEN_L (i.e. normal load) as messages per CPU usage per interval (i.e. the inverse value of CPU usage per messages received). Similarly, malicious traffic was explored by running the simulator with only the botnet and backend. From the averaged interval values, we extracted the mean DDoS message count and combined this with mean CPU usage per interval to estimate $DDoS_L$ (i.e. attack load). These estimates were applied to the dynamical DDoS model to generate values for ω and γ.

We then ran the full simulation (with clients, PLCs, the backend, and the botnet) in FET-SER mode for multiple runs of around 5 minutes each. Using the averaged interval values, the mean benign and attack message arrival rates were used to estimate BEN_A and $DDoS_A$.

Conceptually, the server-database backend combination is aligned to the target network T in the F-C model. Based on observations of CPU usages spikes and correlated timeouts, we defined a functionality threshold for the server and determined its status (F or C) by comparing the threshold to its CPU readings over time. The wider management network of PLCs (which depend on T) is then aligned to network Y. A PLC's functionality is a function of the availability of its prerequisite node, which here is the server. Therefore, another threshold was selected for timeouts received for forwarded batch registration requests. When timeouts exceed the threshold, we classify the dependency link as failed and subsequently, the PLC as compromised. Finally, the client layer is aligned to the dependent network X. Once again, clients fail when their dependency links fail. Client population status is hence defined by the total number of failed registration attempts logged on all PLCs in the aggregation layer.

Some minor adjustments were made to the models to better align with the simulator scenarios. Firstly, a birth rate μ was added to the F-C model to account for the simulator's steady addition of new smart meters. Where p is the sampled number of new nodes added, network Y's equations are updated as follows:



(a) AMI simulation environment. (b) F-C model (single run). (c) F-C model (mean).

Figure 7: Comparison of compromised population results generated by the simulator and predicted by the F-C model.

$$P_{Y1} = \mu \qquad \{F_Y + p, C_Y + 0\} \qquad (26)$$

$$P_{Y2} = \omega_D\,(\lambda_{YX}\,F_Y\,C_X) \qquad \{F_Y - 1, C_Y + 1\} \qquad (27)$$

$$P_{Y3} = \omega_D\,(\lambda_{YY}\,F_Y\,C_Y) \qquad \{F_Y - 1, C_Y + 1\} \qquad (28)$$

$$P_{Y4} = \gamma C_Y \qquad \{F_Y + 1, C_Y - 1\} \qquad (29)$$

Meanwhile, to account for the variability observed in CPU usage on the server, the dynamical DDoS model was updated so that traffic loads (BEN_L and $DDoS_L$) are sampled from the normal distribution using the estimates described previously. Similarly, to account for the variation in traffic arrival rates at the server (due to possible losses during transmission), the arrival rates (BEN_A and $DDoS_A$) are sampled in the same way. The Python scripts use the `psutil` package, which logs CPU usage as a percentage over the number of available CPU cores each time it is called. Therefore, the value of M was kept consistent with the numerical tests, and $m(t)$ was converted into a percentage of the total available M consumed, allowing direct comparison. Lastly, α was lowered to 50 based on log observations.

4.2.2. Results

We first note some general trends observed in the simulation environment to assess the behavioural assumptions made when developing the models. Mean resource loads for benign (BEN_L) and DDoS ($DDoS_L$) messages (under our estimations) were relatively close but with DDoS messages having slightly less load. For example, in a single round, BEN_L was estimated at 11.32, whilst $DDoS_L$ was estimated at 7.23. This makes sense as we would expect DDoS messages to aim to mimic normal behaviour in order to stay inconspicuous for as long as possible. Keeping message size slightly smaller can enable this, balancing the malicious message arrival rates which are, of course, far larger. For example, in the same round, we estimated BEN_A to be 4.59, whilst $DDoS_A$ was 77.66 messages per interval.

Figure 6 shows the resource consumption, or the inverse $m(t)$, recorded by the simulator (a), and generated by the updated dynamical DDoS model for a single run (b) and averaged over multiple runs (c). They show relatively manageable consumption caused by benign traffic, with irregular spikes and some high peaks. A steadily increasing trend can be seen in the highest spikes, with the simulation outputs portraying a more consistent pattern. When compared against the logs, the peaks in (a) correlate with the periodic batch updates pushed by PLCs to the backend. A similarly consistent pattern is therefore not present in the randomised outputs of the generalised DDoS model. Nevertheless, the model provides a close approximation of the simulated activity, especially when averaged. The increasing trend in (a) can be attributed to the growing number of meters joining the network, which

is captured indirectly in (c) as an accumulating load on the targeted resources.

A rapid increase is observed in resource consumption by DDoS traffic in the initial stages of the observation. In (a), consumption remains consistently high, but fluctuates around the estimated maximum (as split across multiple CPUs), before starting to fall as the simulation ends. For the model-generated results, we see DDoS traffic maintain a similarly high consumption rate, but due to our flat approximations, without the same fluctuations. The small changes in the simulation logs may be caused by variable processing delays on the server, which the current model does not consider. However, despite these simplifications, the dynamical DDoS model still appears to capture the key contributing factors and the correct trajectories for attack impact. We have also demonstrated that fine-tuning and adaptation is possible to fit the model to particular scenarios, which provides a good foundation for further development.

Figure 7 shows the compromised populations for a sample test case as generated by the simulator (a), and as predicted by the *F*-*C* model for a single run (b) and averaged across multiple runs (c). Between the simulated and model-generated results, we can see a comparable pattern in C_T, with initial fluctuations that eventually settle into a constant state of compromise for the rest of the observation period. This is because in the simulator, the server quickly begins to suffer from high CPU usage as the load of the DDoS attack accumulates, which corresponds with the rate of depletion on the representative T node in the model.

The predicted population trajectories for C_X are also closely matched. In the simulated output, C_X continues to fluctuate, whereas in the model output, it becomes constant. Investigations of the logs suggest that this is caused by the variation in the requests each PLC receives per interval. Since PLC status per interval depends on the number of timeouts received, periods of fewer incoming requests lead to corresponding periods with fewer timeouts. If under the defined threshold, this leads to that PLC being recorded as functional. Meanwhile, in the *F*-*C* model, we assume a constant level of dependency, which leads to less fluctuation between F and C states. Otherwise, similar to the server, the PLCs initially vacillate a little between F and C, before at least 2 out of the 3 (on average) become compromised and remain so for most of the observation period.

The C_Y population generally shows rapid increase caused by the gradual addition of new nodes to the client layer. During a simulation run, newly added smart meters will be recorded as compromised when they try to register, such that the time to failure will be the delay between the meter's creation and the PLC's failed batch forward request attempt to the server. With a short forwarding interval and an ongoing attack, this happens quickly, causing a sharp increase in C_Y. Hence, we found that the value of p in relation to the birth rate μ must be sufficiently high to match this, and variation in meters added per interval should be accounted for. This was easily tuned by sampling around an estimated mean value.

We also noted during these tests that a limitation of the model is that for each tier, the initial starting population must be greater than 0 to ensure some active rates of change. We therefore initialised the model F_T=1, F_X=3, and F_Y=2, assuming 2 existing smart meters instead of 0 as in the simulator. Nevertheless, taken alongside the numerical results, the model appears to show good performance for larger and smaller populations.

Overall, the results show the *F*-*C* model performs well in approximating the compromise rates and patterns of interconnected networks under a DDoS-type attack. Despite some minor discrepancies caused mostly by the level of abstraction necessitated by this type of model, the simulation comparisons show that the population-based approach to exploring interdependency has validity and is a plausible alternative to graph-based methods.

5. Discussion

As discussed in Section 2, many existing works use graph-based percolation and/or simulation-style modelling. The main advantages of this are the high-level of detail that can be considered, including individual node and link behaviours, and the application of graph theory for analysis. In contrast, the *F*-*C* model follows the epidemiological modelling approach to observe systems at the population level. It is designed to estimate the scale of an incident by measuring rates of change in populations, giving a high-level view of the situation rather than details of specific node and edge failures. This enables the tracking of node statuses in large populations that would

otherwise be difficult and costly to model. Given the advantages of both approaches, the intention is for the *F-C* model to sit alongside graph-based methods to provide a simpler and quicker (and perhaps in some cases an interim) way to estimate network vulnerability.

The results presented in the previous section show that this is possible, as both *F-C* and DDoS models were able to closely approximate the behaviours of the AMI simulator. Furthermore, the *F-C* model makes it easier to test different structures, without having to model them at full-scale, simply by adjusting the dependency parameters and edge probabilities. The numerical analyses showed that the intra-dependency of a network can amplify its inter-dependency with another network, assuming that inter-dependencies with the latter are required for the fulfilment of intra-dependencies in the former. Hence, by exploring different network structures, it may be possible to optimise towards maximum functionality within a threshold for interdependency. Alternatively, different structures may be explored to add network redundancy [32]. This is a topic for further research.

Another avenue for research is in deriving formulations for ω_D, which was shown to be distinct from attack-driven depletion (ω_A), and important in driving subsequent compromises. It should be a function of the type of dependency that exists between two nodes or populations, considering what it delivers (e.g. data, power, or access) and the direction. For example, Dudenhoeffer [33] included categories for physical, informational, and procedural, whilst Falahati et al. [34] defined direct and indirect component-to-component, and network-to-component dependencies. Additionally, the numerical tests showed that E_X, E_Y, and E_Z scaled the levels of compromise in and across networks. Therefore, the quantity of each type of dependency must also be considered.

A disadvantage of the epidemiological approach is that it requires a level of abstraction and generalisation of individual characteristics within a defined population. This assumption of homogeneity can be mitigated by grouping similar devices within the same logical or geographical area into separate sub-populations. We posit that it can be reasonably assumed that for a single deployment scenario by a single organisation, devices serving similar functions are likely to have a similar make, model, capabilities, and constraints. These details can then be the basis for defining dependencies and failure thresholds, thereby preserving some node heterogeneity.

Another potential disadvantage is that the model reveals the numbers of *F* and *C* nodes, but does not reveal where surviving node clusters or giant components may lie. The use of sub-populations can also help with this, especially if multiple smaller groups are included. In this case, a mostly-functional sub-population at the end of the test would represent the surviving (and/or recovered) nodes. Furthermore, the 3-tiered version of F-C can be expanded by adding a new population and defining its connectivity. This could be used to explore larger or more complex smart grid networks, as well as external systems with which the grid might have interdependent relationships [27].

In future iterations, we aim to further integrate the *F-C* and DDoS models so that they run simultaneously and so variable depletion and recovery rates can be considered. This will allow node-specific behaviours, like the request transmission intervals of the simulated devices, to be included so that fluctuations in activity are better represented. The simulations also highlighted the role of transmission channels which may cause delays or otherwise affect the arrival rate of messages on the target. Enabling the use of variable depletion rates would mean arrival rates can be applied stochastically per interval.

Finally, it is of course possible that the smart grid is targeted with attacks other than DDoS, and may be even be confronted with multiple types of attack simultaneously in a single campaign [27]. The design of the *F-C* model means that ω_A can theoretically represent the degradation in functionality caused by any type of attack, single or combined. The complexity associated with the definition of ω_A will be offloaded to an attack model (like the DDoS one presented in this work), making the *F-C* model flexible in this regard. Another possibility is the exploration of different types of DDoS attack, including low-rate, with variation of the DDoS model. The previously mentioned integration approach should therefore be simple enough to switch out attack models for this purpose.

6. Conclusions

Society's growing demand for electricity has driven the move to smart grids, which integrate IT systems with the

existing power infrastructure. This introduces interdependencies between communication and power networks, so that disruptions at one point can propagate iteratively throughout the entire grid. Furthermore, communication networks introduce cyber security vulnerabilities to the previously private power grid systems. Components and flows are vulnerable to DDoS attacks that can deplete the resources of a link or a device, causing it to fail in its duties to its dependent nodes, triggering a cascade of disruption. It is therefore important to identify interdependency relationships and the trajectories of such cascading failures so that better defence and contingency approaches can be developed.

To address this, we have presented the epidemiologically-based *F-C* compromise propagation model, supported by the dynamical DDoS model, to explore DDoS attacks on a targeted (Internet-facing) portion of the IT network and the subsequent impact this has on attached dependent OT network. Through numerical testing and multivariate analysis, we explored the impact of the defined parameters, and validated the models using an AMI simulation environment. The results showed that the proposed models can satisfactorily estimate the simulator results, and can therefore be considered in future as an alternative to graph-based modelling for predicting the vulnerabilities of interconnected networks, such as those within critical infrastructure like smart grids.

Acknowledgements

This work is funded by and a part of Energy Shield, a project under the European Union's H2020 Research and Innovation Programme (Grant No: 832907).

References

[1] J. Momoh, Interoperability, Standards, and Cyber Security, 2012, pp. 160–175. doi:10.1002/9781118156117.ch8.

[2] A. Gopstein, C. Nguyen, C. O'Fallon, D. Wollman, N. Hasting, NIST Framework and Roadmap for Smart Grid Interoperability Standards, Release 4.0 (2020).

[3] C.-P. S. P. W. Group, et al., Framework for Cyber-Physical Systems: Volume 1, Overview, Version 1.0, NIST Special Publication (2017) 1500–201.

[4] K. Kimani, V. Oduol, K. Langat, Cyber Security Challenges for IoT-Based Smart Grid Networks, International Journal of Critical Infrastructure Protection 25 (2019) 36–49.

[5] D. Upadhyay, S. Sampalli, SCADA (Supervisory Control and Data Acquisition) Systems: Vulnerability Assessment and Security Recommendations, Computers & Security 89 (2020) 101666.

[6] Y. Mo, T. H.-J. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, B. Sinopoli, Cyber-Physical Security of a Smart Grid Infrastructure, Proceedings of the IEEE 100 (1) (2011) 195–209.

[7] M. Z. Gunduz, R. Das, Cyber-Security on Smart Grid: Threats and Potential Solutions, Computer Networks 169 (2020) 107094.

[8] G. A. Jaafar, S. M. Abdullah, S. Ismail, Review of Recent Detection Methods for HTTP DDoS Attack, Journal of Computer Networks and Communications 2019 (2019).

[9] Energy Shield, last accessed 10th February 2021 (2021).
URL https://energy-shield.eu/

[10] D. Acarali, M. Rajarajan, D. Chema, M. Ginzburg, Modelling DoS Attacks & Interoperability in the Smart Grid, in: 2020 29th International Conference on Computer Communications and Networks (ICCCN), IEEE, 2020, pp. 1–6.

[11] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, S. Havlin, Catastrophic Cascade of Failures in Interdependent Networks, Nature 464 (7291) (2010) 1025–1028.

[12] S. Ruj, A. Pal, Analyzing Cascading Failures in Smart Grids Under Random and Targeted Attacks, in: 2014 IEEE 28th International Conference on Advanced Information Networking and Applications, IEEE, 2014, pp. 226–233.

[13] C. D. Brummitt, R. M. D'Souza, E. A. Leicht, Suppressing Cascades of Load in Interdependent Networks, Proceedings of the National Academy of Sciences 109 (12) (2012) E680–E689.

[14] V. Rosato, L. Issacharoff, F. Tiriticco, S. Meloni, S. Porcellinis, R. Setola, Modelling Interdependent Infrastructures Using Interacting Dynamical Models, International Journal of Critical Infrastructures 4 (1-2) (2008) 63–79.

[15] V. Rosato, L. Issacharoff, S. Meloni, D. Caligiore, F. Tiriticco, Is the Topology of the Internet Network Really Fit to Sustain its Function, Physica A: Statistical Mechanics and its Applications 387 (7) (2008) 1689–1704.

[16] Y. Cai, Y. Cao, Y. Li, T. Huang, B. Zhou, Cascading Failure Analysis Considering Interaction Between Power Grids and Communication Networks, IEEE Transactions on Smart Grid 7 (1) (2015) 530–538.

[17] C. Poulin, M. Kane, Identifying Heterogeneous Infrastructure Interdependencies through Multiverse Simulation, in: 2019 Resilience Week (RWS), Vol. 1, IEEE, 2019, pp. 123–131.

[18] D. Dagon, C. C. Zou, W. Lee, Modeling Botnet Propagation Using Time Zones, in: NDSS, Vol. 6, 2006, pp. 2–13.

[19] C. C. Zou, W. Gong, D. Towsley, Code Red Worm Propagation Modeling and Analysis, in: Proceedings of the 9th ACM conference on Computer and communications Security, 2002, pp. 138–147.

[20] F. Wang, Y. Zhang, C. Wang, J. Ma, S. Moon, Stability Analysis of a SEIQV Epidemic Model for Rapid Spreading Worms, Computers & Security 29 (4) (2010) 410–418.

[21] E. Kenah, J. C. Miller, Epidemic Percolation Networks, Epidemic Outcomes, and Interventions, Interdisciplinary Perspectives on Infectious Diseases 2011 (2011).

[22] S. Y. Del Valle, J. M. Hyman, N. Chitnis, Mathematical Models of Contact Patterns Between Age Groups for Predicting the Spread of Infectious Diseases, Mathematical Biosciences and Engineering: MBE 10 (2013) 1475.

[23] S. Ramanauskaite, A. Cenys, Composite DoS Attack Model - Jungtinis DoS Atakku Modelis, Mokslas - Lietuvos ateitis/Science - Future of Lithuania 4 (1) (2012) 20–26.

[24] K. Johnson Singh, T. De, Mathematical Modelling of DDoS Attack and Detection Using Correlation, Journal of Cyber Security Technology 1 (3-4) (2017) 175–186.

[25] K. Shuaib, Z. Trabelsi, M. Abed-Hafez, A. Gaouda, M. Alahmad, Resiliency of Smart Power Meters to Common Security Attacks, Procedia Computer Science 52 (2015) 145–152.

[26] J. Cao, W. S. Cleveland, D. X. Sun, Bandwidth Estimation for Best-Effort Internet Traffic, Statistical Science (2004) 518–543.

[27] L. Das, S. Munikoti, B. Natarajan, B. Srinivasan, Measuring Smart Grid Resilience: Methods, Challenges and Opportunities, Renewable and Sustainable Energy Reviews 130 (2020) 109918.

[28] M. Janić, Modelling the Resilience of Rail Passenger Transport Networks Affected by Large-Scale Disruptive Events: The Case of HSR (High Speed Rail), Transportation 45 (4) (2018) 1101–1137.

[29] E. B. Wilson, J. Worcester, The Law of Mass Action in Epidemiology, Proceedings of the National Academy of Sciences of the United States of America 31 (1) (1945) 24.

[30] L7 Defense, last accessed 10th February 2021 (2021).
URL `https://www.l7defense.com/`

[31] V. Kocabas, S. Dragicevic, Assessing Cellular Automata Model Behaviour Using a Sensitivity Analysis Approach, Computers, Environment and Urban Systems 30 (6) (2006) 921–953.

[32] G. Fu, R. Dawson, M. Khoury, S. Bullock, Interdependent Networks: Vulnerability Analysis and

Strategies to Limit Cascading Failure, The European Physical Journal B 87 (7) (2014) 1–10.

[33] D. D. Dudenhoeffer, M. R. Permann, M. Manic, CIMS: A Framework for Infrastructure Interdependency Modeling and Analysis, in: Proceedings of the 2006 Winter Simulation Conference, IEEE, 2006, pp. 478–485.

[34] B. Falahati, Y. Fu, L. Wu, Reliability Assessment of Smart Grid Considering Direct Cyber-Power Interdependencies, IEEE Transactions on Smart Grid 3 (3) (2012) 1515–1524.

Dilara Acarali is a researcher at the Institute for Cyber Security at City, University of London, where she earned a PhD in Information Engineering. She also holds a BEng in Internet Systems Engineering. Her current research is focused on the predictive modelling of cyber attacks, and her wider research interests include data privacy, data analytics, IoT security, and sustainable technology. She has industry experience in network engineering and network security, having worked many years within the UK technology sector for telecoms and IT service providers.

K. Rajesh Rao received a B.E. degree in Computer Science and Engineering and an M.Tech. degree in Computer Science and Information Security. His Ph.D. degree is in the area of Information Security from Manipal Academy of Higher Education (MAHE), Manipal, India. Currently, he is an Assistant Professor-Senior at Manipal Institute of Technology, MAHE, and is also associated with City, University of London as a Researcher in the area of cyber security. His research interests include, but are not limited to, security analytics, access control models, cloud security, internet of things, and soft computing.

Muttukrishnan Rajarajan is Professor of Security Engineering at the City, University of London, where he currently leads the Institute for Cyber Security. He is a Visiting Researcher with British Telecom's Security Research and Innovation Laboratory. His research interests include privacy-preserving data analytics, cloud computing, IoT security, and wireless networks. He has published well over 300 articles and continues to be involved in the editorial boards and technical programme committees of several international security and privacy conferences and journals. He is an Advisory Board Member of the Institute of Information Security Professionals, U.K., and an advisor to the U.K. Government's Identity Assurance Programme (Verify U.K.).

Doron Chema is a co-founder of L7 Defense who has played the role of CEO from its day of establishment. Doron came up with the L7 innovative platform technology idea in the early days, and has since managed efforts in its on-going development into a full-scale product and solution suite, side-by-side with the management of the company. Doron has a vast amount of experience in customer-facing product development, as well as in architecture design, bio-algorithms, and enterprise software development. Prior to L7 Defense, he led major R&D efforts in various positions in the Israeli hi-tech industry. Doron holds a Ph.D. degree in Bioinformatics from the TLV University.

Mark Ginzburg is a co-founder of L7 Defense who has played the role of Head of R&D and algorithms from its day of establishment. He has a vast amount of hands-on and management experience in various fields of technology and algorithms, including ML/AI, cloud architecture, cyber security, cryptography, and computer vision. Prior to L7 Defense, Mark managed algorithm research and development teams in the Israeli hi-tech industry and military. Mark holds an MSc degree in Computer Science from the Technion (IIT).