A Model Driven Approach for Assessing Survivability Requirements of Critical Infrastructures

Ugo GENTILE [a,e] and Simona BERNARDI [b] and Stefano MARRONE [c,1] and José MERSEGUER [d] and Valeria VITTORINI [a]

[a] *DIETI, Università degli Studi di Napoli "Federico II", Napoli (Italy)*
[b] *DMF, Università della Campania "Luigi Vanvitelli", Caserta (Italy)*
[c] *CUD, Academia General Militar, Zaragoza (Spain)*
[d] *DIIS, Universidad de Zaragoza, Zaragoza (Spain)*
[e] *Engineering Department CERN, CH-1211. Geneva 23 (Switzerland)*

Abstract. Critical infrastructures are complex networked systems. They must be able to provide essential services, even when they are compromised by intentional or accidental threats. Guaranteeing essential services means to ensure survivability with an adequate Quality of Service (QoS). This paper proposes a model-driven approach for the assessment of survivability requirements. In particular, we propose a graphical Survivability Assessment Model (SAM), based on UML. It is automatically derived from a UML specification that encompasses essential services, service modes, threats and survivability strategies. Furthermore, model-driven techniques are used to assess the SAM. Then, we propose some preliminary property verifications to discover flaws in the specification. The model driven paradigm is used to ensure high level of usability and abstraction of the artifacts that are key issues in communication among stakeholders. The approach has been applied to a scaled-down model of a smart grid, an evolution of traditional power grids based on high performance and dependable computer networks.

Keywords. Security Requirement Elicitation, UML Misuse Case, Service Survivability, Model Transformations, Networked Systems Security

1. Introduction

The integration of computer networks within complex critical infrastructures introduces new threats and technological issues. They can compromise the infrastructure correct functioning and also harm human life. An example of critical infrastructure is constituted by smart energy grids. These are computing and communication infrastructures used to gather information about suppliers and con-

[1] Corresponding Author: Stefano Marrone, Dipartimento di Matematica e Fisica, Università della Campania "Luigi Vanvitelli", viale Lincoln 5, 81100 Caserta, Italy; E-mail: stefano.marrone@unicampania.it

sumers. The aim is for optimizing the energy supply, then improving the overall efficiency, reliability, and sustainability in an autonomic way. In this context, it is a crucial need to ensure service survivability; this is *the capability of a system to provide "essential services" with a specified Quality of Service despite the occurrence of threats or attacks* [6].

Eliciting survivability requirements in such infrastructures can be a hard task. Consider that the earliest phases of system development are the most critical. This is because an incomplete and/or inconsistent requirement specification can bring to vulnerable systems. We consider models as first class citizens in the assessment of requirements for critical systems. Then, we propose a Survivability Assessment Model (SAM), that can be formally verified. It will be an important support for requirement engineers in identifying gaps and/or errors in the system specification.

In this paper, we then propose an approach to generate a SAM using model-driven techniques. The approach consists of two steps. First, an improved Misuse Case Diagram (MUCD) specification is produced. It considers system essential services, service modes, threats and mitigation strategies. Second, a state-based SAM is automatically generated from the MUCD specification, by means of model-driven techniques. The SAM enables engineers to easily discover flaws in the system specification.

This work builds on the results presented in [3], where a Petri Net model was generated from the SAM to verify survivability properties through model checking. The original contribution of this paper is to support the verification of such properties directly on the SAM by means of model queries without generating state-based models that scale hard with the dimension of the system.

The proposed approach is applied for the survivability assessment of a scaled-down model of a smart grid.

The structure of the paper is the following. Section 2 introduces the Smart Grid case study. Section 3 overviews the proposed approach. Section 4 describes the system specification process and applies it to the Smart Grid case study. Section 5 focuses on the verification process of SAM survivability properties. Section 6 provides a review of the related literature. Finally, Section 7 provides closing remarks.

2. The Smart Micro Grid case study

Traditional power grids rely on dated architecture where the generation is centralized — mostly based on non-renewable sources. Moreover, the energy is delivered to passive consumers through the transmission and distribution lines [14]. Very often they are affected by blackout events, especially during peak energy periods (e.g., [9]), cascading failures (e.g., [12]), energy lacks and thefts along the distribution line. On the other hand, they are also a sensitive target for terrorist attacks (e.g., [1]). Such limitations clearly show that traditional power grids are not able to cope with a higher energy demand, higher security and reliability, lower environmental impacts and reduced operating costs.

To face such issues, several governments and energy agencies are promoting a radical change in electrical distribution on the grid to close energy consumers and consumers. Distributed Energy Resources (DERs) encompass new assets like, e.g, renewable sources such as wind, hydro and/or Solar power. In particular, Distributed Generators or Energy Storages (EGs) that absorb energy peaks and avoid lacks of energy. Finally, the demand-response management policy enables to define a set of modifications in final customers electricity usage, in response to changes in the price of energy or to reduce the energy consumption when the system reliability is compromised [26]. DERs enable customers to take energy from the distribution grid but also to generate energy themselves to satisfy their needs and, possibly, to sell the surplus of energy back to the distribution operator.

The management of such new power grids requires an advanced Information and communication technology (ICT) infrastructure that is able to provide monitoring and measurement functionalities; the effects are to detect incorrect behaviors and failures, to enhance a QoS-aware consumption and to provide billing and accounting functionalities [7, 19]. Security is also a primary issue but it has been traditionally coped by experimental approaches — e.g., PMUs [5], other monitoring technologies [20], big data analysis [30].

The DER approach, currently addressed by several European governments, consists in clustering the whole distribution grid in smaller areas, namely *Smart Micro Grids* [17]. Figure 1 shows a typical architecture, which has been defined considering the requirements collected from surveys [28] and [2]. Electricity consumers (e.g., private houses, public or commercial enterprises) are connected to the distribution lines. Some, or even all, of them own a local power generator to satisfy the local demand. Moreover, they are equipped with a *smart meter*, a device that periodically records the consumption of energy and transmits data to the distribution system operator for metering and billing purposes. Along the

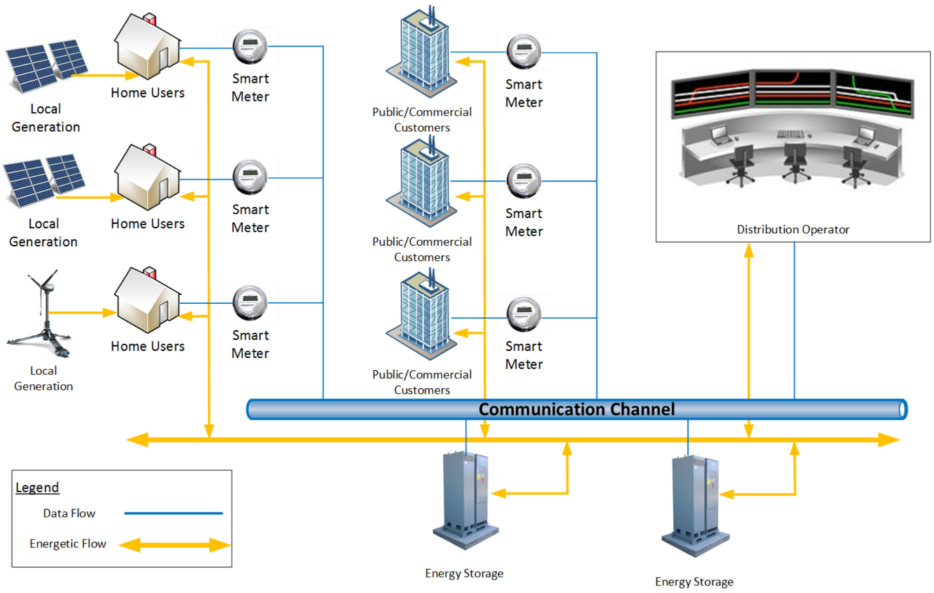

Figure 1. Smart Micro Grid Architecture

distribution lines EGs gather all the energy that has been locally generated but not consumed. Smart Micro Grids provide several advantages for both customer and distribution operator. Customers can (*i*) receive invoices on real consumption, (*ii*) issue tailored tariffs, and (*iii*) obtain savings in billing. Operators can (*i*) obtain peak shaving, (*ii*) realize energy efficiency and CO_2 reduction, (*iii*) reduce commercial and technical losses.

In a more futuristic vision, the appliances connected to the grid will get smarter. For example, to perform more efficient and productive use of electricity (e.g.,reducing power consumption during peak hours and operating during those hours when power costs less). Also they will have two-way communication links, allowing commands to be sent toward the smart appliances for multiple purposes as for example the remote service disconnections.

3. Approach overview

The approach focuses on the survivability assessment of the system. In particular, the aim is to produce an improved system requirement specification and to leverage it for verifying system survivability properties. Figure 2 provides an overview of the approach by highlighting the main tasks and produced artifacts — specifications or models — which are described in the following.

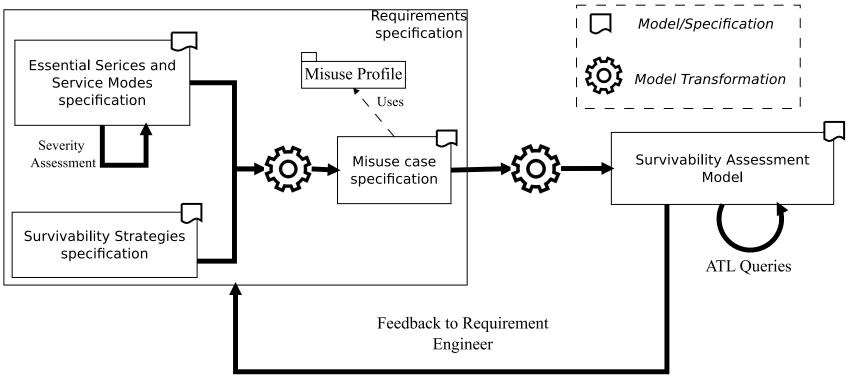
Requirements specification
Essential Serices and Service Modes specification
Severity Assessment
Misuse Profile
Uses
Misuse case specification
Survivability Strategies specification
Model/Specification
Model Transformation
Survivability Assessment Model
ATL Queries
Feedback to Requirement Engineer

Figure 2. Approach overview.

1. *Essential Services and Service Modes specification.* During the requirements elicitation stage, the engineer identifies the system *essential services*, i.e., services that must operate even when affected by faults or attacks. Also the engineer identifies *QoS metrics* related to services, which lead to the definition of the system *service modes.*
2. *Severity assessment.* This activity is aimed to grade the service modes by their *severity.* We conform to the severity concept adopted by Symantec [24] where higher is the severity, higher the damage in case the service mode is entered.

3. *Survivability strategies specification.* During the requirements elicitation, potential threats to the system and related countermeasures — i.e., resistance, recognition and recovery strategies [6] — need to be identified.
4. *Misuse case specification.* Using the the Misuse Case UML profile [22] the specifications produced in the previous steps are used to generate a misuse case specification (MUCD) using a text-to-model (T2M) transformation.
5. *Survivability Assessment Model.* The misuse case specification is used to generate, by means of a model-to-model (M2M) transformation, the survivability assessment model (SAM). The SAM is a UML state machine model.
6. *Survivability Properties Verification.* This activity is aimed to verify some predefined system survivability properties by submitting specific queries on the SAM. The results of this activity provide a feedback to the engineer, enabling him/her to discover non correct and non complete points in the misuse case specification — such as attacks that have not been addressed by a proper recovery.

4. Requirements specification

This section describes the requirements elicitation process that enables to produce the misuse case specification (MUCD) of the system. The Smart Micro Grid, introduced in Section 2, is used as running example.

4.1. Essential Services and Service Modes specification

An initial task consists in identifying the system service modes, that are defined in terms of the Quality of Service (QoS) of the essential services. This phase produces a table (Table 1) that specifies the service modes of a Smart Micro Grid ICT infrastructure. Among the services offered by the system, the engineer — possibly with the help of domain experts — selects those that need to survive despite faults or attacks. For our case study, four essential services are identified [28]. Two of these services concern the exchange of information between a smart meter and a distribution system operator: they are the request of energy prices (*Ask Energy Prices*) and the transmission of customer's account and balance information (*Report Account and Balance*). The other two services are provided by the distribution operator: they are related to the estimation of the current state of the grid (*Estimate State*) and the management of energy production and consumption (*Demand Response*).

The QoS of an essential service is defined by assigning acceptable threshold values to a subset of metrics of interest. For example, three metrics have been defined for the case study: the steady state availability ($ssAvail$), the confidentiality level ($confLevel$) and the integrity level ($integLevel$). Table 1 — row labelled *FF* — specifies the *Full Functionality* system service mode that represents the best QoS offered by the system [21, 25] for the identified essential services. Observe that essential services can be characterised by different threshold values and all the metrics values express required minimum thresholds. The other rows of the table, i.e., *Degraded Integrity* (DI), *Degraded Confidentiality* (DC), *Degraded Availability* (DA) and *Maximum Degradation* (MD), specify different system service modes considering possible — although still acceptable — degraded values of (a subset of) the QoS metrics.

Severity assessment Once the system service modes have been defined, they are ranked. The rank considers the relevance of the degraded QoS metrics and their minimum threshold values. Therefore, we can provide different priorities for intervention, just in case the system is threatened by attacks. In the case study, the steady state availability is considered the QoS metric of major importance followed by the confidentiality level. The integrity level is instead the QoS metric with minor relevance. Table 1 — second column — shows the severity level associated to each degraded service mode of the case study, where the lowest level corresponds to the highest degradation in compliance with Symantec [24]. In particular, the *DI* service mode corresponds to a minor degradation. This is because only the integrity level, i.e., the QoS metric with minor relevance, has a threshold value that is degraded with respect to the one in the fully operational service mode (i.e., *FF). The* DC *service mode has lower severity level (i.e.,* L3*) than the previous mode since the confidentiality level, i.e., the QoS metric of medium relevance, is degraded. The* DA *service mode is characterised by the degradation of the steady state availability, that is the QoS metric of major relevance, then it is more severe than the* DC *mode. Finally the* MD *service mode has the lowest severity level, since the steady state availability is even more degraded.*

			Essential services			
	Severity level	QoS metric	Ask Energy Prices	Report Account and Balance	Estimate State	Demand Response
FF	-	ssAvail	-	-	(99.99%,min)	(99.99%,min)
		confLevel	(medium,min)	(medium,min)	(high,min)	-
		integLevel	-	(medium,min)	(high,min)	(high,min)
DI	L4	ssAvail	-	-	(99.99%,min)	(99.99%,min)
		confLevel	(medium,min)	(medium,min)	(high,min)	-
		integLevel	-	(low,min)	(medium,min)	(medium,min)
DC	L3	ssAvail	-	-	(99.99%,min)	(99.99%,min)
		confLevel	(low,min)	(low,min)	(medium,min)	-
		integLevel	-	(low,min)	(medium,min)	(medium,min)
DA	L2	ssAvail	-	-	(99.9%,min)	(99.9%,min)
		confLevel	(low,min)	(low,min)	(medium,min)	-
		integLevel	-	(low,min)	(medium,min)	(medium,min)
MD	L1	ssAvail	-	-	(99.0%,min)	(99.0%,min)
		confLevel	(low,min)	(low,min)	(medium,min)	-
		integLevel	-	(low,min)	(medium,min)	(medium,min)

Table 1. Specification of severity levels and QoS metric values for each service mode.

4.2. Survivability strategies specification

According to [6], once we have identified the essential services we need to carry out two tasks: an intrusion analysis *based on the system environment; and a* survivability analysis *to identify key countermeasures that may include* resistance, recognition *and* recovery *strategies [6].*

Intrusion analysis *identifies threats to essential services. This is a process carried out by an expert, basically for each threat he/she analyses the consequences for* *the system in terms of the degradation that occurs. We summarise the intrusion analysis through a table (e.g., Table 2).*

Survivability analysis *is a process carried out by a domain expert, who identifies for each threat countermeasures of resistance, recognition or recovery. We summarise the survivability analysis through a table (Table 3).*

In the following, we report the intrusion and survivability analyses for the case study. An ICT infrastructure of a Smart Micro Grid can be threaten by attacks against: a) the availability*, to disrupt the energy delivery; b) the* confidentiality*, to stole information from customers and/or the distributor operator; and c) the* integrity*, to introduce fake commands that can compromise the grid. In particular, we identified the following threats (see Table 2):*

- Data Injection *is an attack to disrupt* integrity*, and it mostly threats the* Report Account and Balance, Estimate State *and* Demand Response *essential services.*
- Spoofing *threats* confidentiality *through the* Report Account and Balance *and* Estimate State *services.*
- Eavesdropping *threats* confidentiality *of the Smart Grid, essentially through the* Ask Energy Prices *service.*
- Flooding *threats* availability *of the Grid by sending a large number of packets. The services exposed are* Estimate State *and* Demand Response*.*
- Electromagnetic Interference (EMI) *is a disturbance of the communication channel that can degrade or totally disrupt the service. The source can be a specific device (malicious attack) or a natural phenomena (accidental threats). The* Demand Response *is the target service.*

	Threats				
Essential services	**Data Injection**	**Spoofing**	**Eavesdropping**	**Flooding**	**EMI**
Ask Energy Prices			DC		
Report Acc. & Bal.	DI	DC			
Estimate State	DI	DC		DA	
Demand Response	DI			DA	MD

DI: Degraded Integrity; DC: Degraded Confidentiality;
DA: Degraded Availability; MD: Max. Degradation

Table 2. Summary of the Smart Grid intrusion analysis.

Concerning the survivability analysis*, we relied on the work of Wang and Lu [27] that identifies several countermeasures. We classified them according to the survivability strategy terminology — introduced by [6] — as follows:*

- Resistance *is the capability to repel attacks and allows the system to remain in full functionality [6]. The analysis devises Encryption (RES1) and Point-to-Point Authentication (RES2). The former mitigates the threats against confidentiality and the latter those against integrity.*
- Recognition *is the capability to detect attacks as they occur and to evaluate the extent of damage and compromise [6]. The analysis devises: Signal-based detection (DET1) for EMI; Packet-based detection (DET2) for flooding;*

and Anomaly-based network detection (DET3) for Data Injection, Spoofing and Eavesdropping.

- Recovery *is the capability to maintain essential services during attack, limit the extent of damage, and restore services following attack [6]. The analysis devises: a) Rate-limiting (REC1) to mitigate threats again system availability; b) Spread Spectrum protocols (REC2) to mitigate effects due to physical attacks as EMI; and c) Reconfiguration (REC3) to change network topology and reduce the effect of flooding attacks.*

	Strategies							
Threats	RES2	DET3	RES1	DET2	REC1	DET1	REC2	REC3
Data Injection	resistance	recognition						
Spoofing		recognition	resistance					
Eavesdropping		recognition	resistance					
Flooding				recognition	recovery target=FF			
EMI						recognition	recovery target=DA	recovery target=FF

Table 3. Summary of the Smart Grid survivability analysis.

4.3. Misuse case specification

The table-based specification created so far is used to produce an improved misuse case diagram (MUCD). Use cases [10] are a well-known technique for eliciting requirements and likely the most popular. Although they are a suitable technique to capture functional requirements, they are not likewise suited for eliciting non-functional ones, in particular safety and security requirements. To address this lack, misuse cases *were proposed [23] where use case diagrams are extended to encompass the misuse of the system by a hostile actor. Herein, MUCD have been enhanced by adding new extensions (see Table 4 — bold part) in order to use them for the specification of the survivability requirements. The misuse case extensions have been implemented via UML profiling, which is a lightweight approach to extend the UML meta-modeling elements, e.g., the use cases. The meta-modeling approach enables the interoperability between CASE tools (i.e., exchange of MUCD), whereas the table-based specification does not. Moreover, it supports M2M transformations, as the one we propose in Section 5 for survivability as-*

Stereotype	***Description***	***Tags***	***Extended UML metaclass***
misuse	A threat scenario	**targetServiceMode (name,severityLevel)**	Use case
threatens	A threat to a service		Dependency
mitigates	A threat mitigation		Dependency
resistance	**A resistance strategy**		**Use case**
recognition	**A recognition strategy**		**Use case**
recovery	**A recovery strategy**	**targetServiceMode (name)**	**Use case**

Table 4. Misuse case extensions used in the approach.

sessment. In the following, we describe informally the T2M transformation that, provided in input the table-based specification, produces the MUCD of the Smart Micro Grid case study (see Figure 3). The transformation includes the following steps:

1. *Each essential service of the* Intrusion analysis *table (row names of Table 2) is transformed into a use case (left-hand side of Figure 3).*
2. *Each threat of the* Intrusion analysis *table (column names of Table 2) is mapped to a misuse case (right-hand side of Figure 3). Moreover:*
   - *Each misuse case has a tag – i.e.,* `targetServiceMode` *– that indicates the service mode reached as a consequence of the threat. The tagged-value is derived from the table entries corresponding to the mapped threat.*
   - *A* `threatens` *dependency is created between a misuse case and a use case when the table entry corresponding to the mapped threat and essential service is not empty (central part of Figure 3).*
3. *Each strategy of the* survivability analysis *table (Table 3) is mapped to a new use case that is stereotyped according to the table entries corresponding to the mapped strategy (central part of Figure 3). Moreover:*
   - *Each* `recovery` *use case has a tag – i.e.,* `targetServiceMode` *– indicating the service mode reached as a result of the strategy. The tagged-value is derived from the table entries corresponding to the mapped strategy.*
   - *A* `mitigates` *dependency is created between a* `resistance` *(or* `recognition`*) use case and the misuse case it mitigates when the table entry corresponding to the mapped strategy and threat is not empty.*

Finally, the engineer needs to take the following decisions/actions to complete the misuse case diagram:

1. *Identify, in the problem domain, the actors of the use cases and the hostile actors of the misuse cases. In our case study, the actors are: the Smart Meter, the Distributor Operator, the Energy Storage. The hostile actor is the Attacker.*
2. *Reorganize use cases that represent essential services, using inheritance or inclusion if it is convenient (see Figure 3 in our case).*
3. *For each* recovery *use case, indicate the* recognition *use case it extends. The rational is that a recovery process is launched due to a previous threat recognition. For example, see in Figure 3, the recovery strategies* Reconfigure *and* SpreadSpectrumProtocols *extend the recognition strategy* SignalBasedDetection.
4. *Specify both, use cases and misuse cases, using for example the Cockburn template [4].*

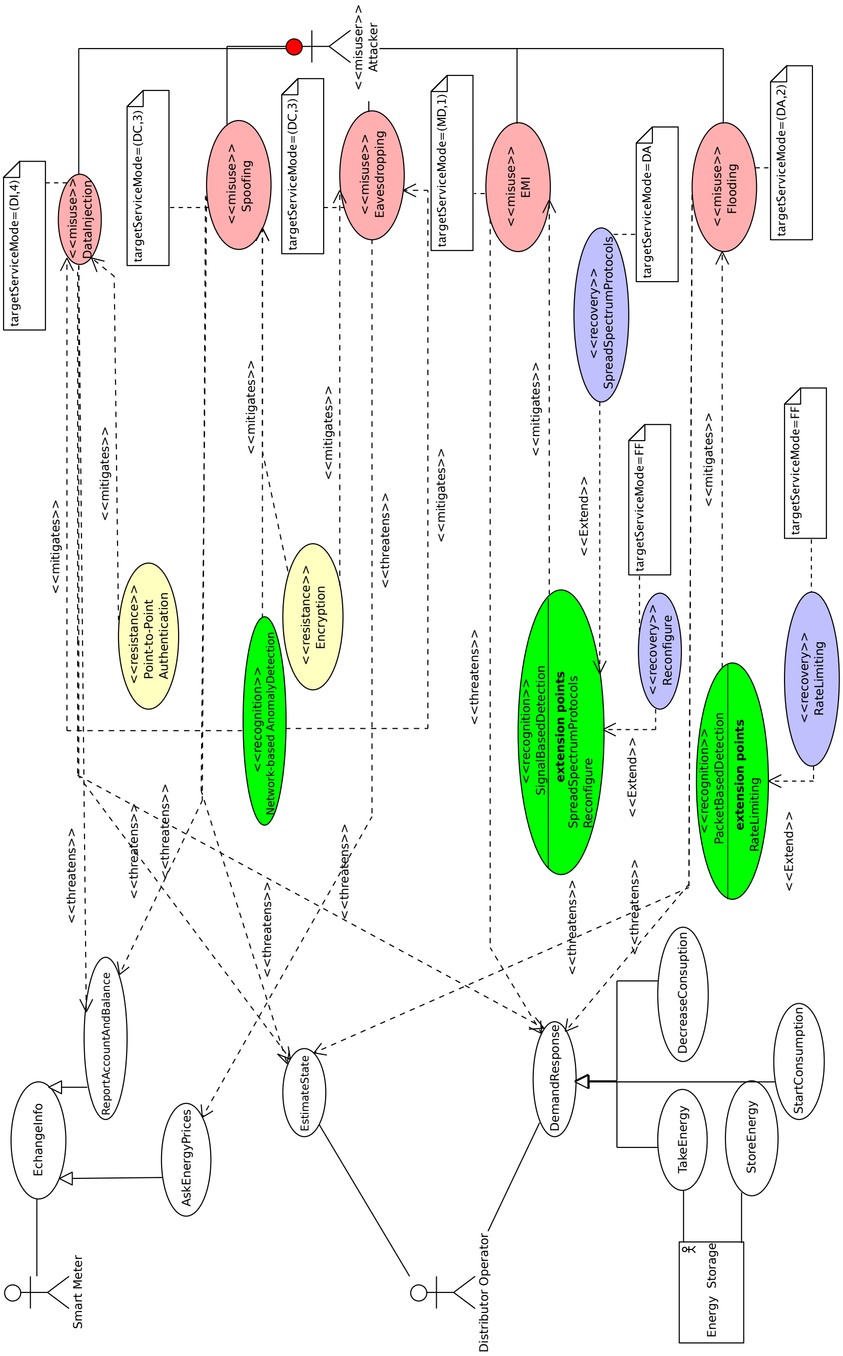

Figure 3. Misuse case diagram of the Smart Micro Grid.

5. Survivability assessment

5.1. Survivability Assessment Model

This section describes the model-driven process that enables to automatically obtain the Survivability Assessment Model (SAM). The SAM is a UML state machine where the states represent the system service modes specified in Subsection 4.1 (see Table 1). Transitions, instead, represent changes of service modes that are triggered by either succeeding attacks or the execution of survivability strategies. The SAM is obtained from the misuse case diagram through a M2M transformation, which has been implemented using the ATL language [13]. The mapping rules of the transformation are described in the following:

- *The main rule generates the UML state machine that contains a* Full Functionality *(*FF*) state, which is stereotyped* `serviceMode`*. Indeed, we assume that every system has always an optimal service mode.*
- *For each misuse case of the MUCD, a new* `serviceMode` *state is generated — if not already created — considering the degraded service mode associated to the misuse case (*`targetServiceMode` *tagged-value). The state is tagged with its corresponding severity level. For the transition generation, two different cases are considered:*
  - ∗ *If the misuse case is mitigated by a* `resistance` *use case, a choice node and three transitions are generated: 1) a transition from the full state and to the choice node, 2) a transition from the choice node back to the full state, which models the succeeding of the resistance strategy, and 3) a transition from the choice node to the degraded state, which models the failure of the resistance.*
  - ∗ *Otherwise, a single transition from the full state to the degraded state is generated.*
- *For each generated service mode* X*, each misuse case is again considered and a transition is created from* X *to the degraded service mode* Y*, indicated by the* `targedServiceMode` *tagged-value associated to the misuse case, only if the severity level of* Y *is less than the one of* X*.*
- *Finally, for each* `recovery` *use case a new transition is added from the degraded service mode — indicated by the tagged-value of the misuse case mitigated by the strategy — to the service mode specified by the* `targetServiceMode` *tagged-value of the use case.*

Figure 4 shows the SAM obtained by the M2M transformation from the MUCD in Figure 3.

5.2. Survivability Properties Verification

The assessment of the MUCD can be performed by verifying survivability properties and providing proper feedback to the engineer about eventual flaws in the system specification. To this aim we use the SAM, obtained via M2M transformation from the MUCD, as input model together with a set of queries that express the

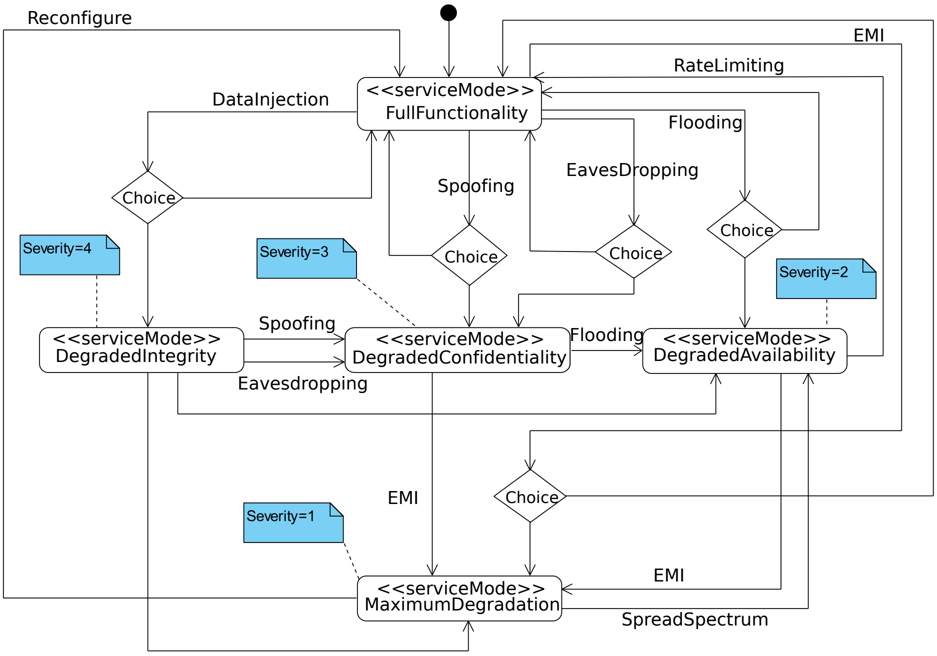

Figure 4. Survivability Assessment Model (SAM)

properties to be verified. Herein, queries are defined by exploiting the mechanisms provided by the ATL language, already used for the M2M transformation.

Considering that the SAM represents the system service modes and the change of service modes due to threats and survivability strategies, it may be interesting to check the following survivability properties:

- ***P1*** *Whenever the threat T occurs, then the survivability strategy S aimed to mitigate T will be executed.*
- ***P2*** *It is always possible to recover the system from a degraded state.*
- ***P3*** *It is always possible to reach the fully operational state from any degraded state.*

The verification of ***P1*** *is quite straightforward since, for each threat T the query has to check if there is an outgoing transition, from the service mode associated to T, that represents the considered survivability strategy S. A more interesting case is represented by the property* ***P2****. In this second case, there is the need to prove that, from each degraded state in the SAM, there is always an outgoing transition which has as target state either the full functionality or a less severe service mode. The ATL query that allows to verify* ***P2*** *is showed in Listing 1. In order to perform the query, several helpers have been defined that are not reported for sake of space.*

Listing 1: ATL query for **P2** verification.

```
query recoverability =
UML!State.allInstances()->iterate(st; state: UML!State =
    thisModule.getFullFunctionality() |
 UML!Transition.allInstances()
 ->select(tr | tr.source = st and tr.source <> thisModule.
    getFullFunctionality() and thisModule.getStateFromName(tr.
    target.name).oclIsTypeOf(UML!State))
 ->select(tra | thisModule.getSeverityFromDegraded(tra.target)>
    thisModule.getSeverityFromDegraded(st)).debug()
);
```

The execution of the query reported in Listing 2, shows that not all the degraded states have at least one outgoing transition directed either to a less severe service mode or to the full functionality state. In particular, such states are the degraded confidentiality state, associated to the Spoofing and Eavesdropping threats, and the degraded integrity state, associated to the DataInjection threat. This feedback is useful since the requirement engineer can correct the specification, introducing proper missing mitigation strategies.

Listing 2: Result of ATL Query for **P2** verification

```
FullFunctionality: Sequence {}
DegradedIntegrity: Sequence {}
DegradedConfidentiality: Sequence {}
MaximumDegradation: Sequence {T_RecoverySpreadSpectrum,
    T_RecoveryReconfigure}
DegradedAvailability: Sequence {T_RecoveryRateLimiting}
```

The ***P3*** property is an example of property that is not possible to verify using the mechanism of the ATL queries. ***P3*** can be easily expressed with temporal logics and, therefore, by translating the property verification problem to a model checking problem. The translation to a model checking problem allows the verification of complex properties and it is currently one of the most relevant future works we are addressing.

6. Related work

The survivability assessment of critical infrastructures is a relevant research trend that, from literature review, appears to be desirable at early stages of system development. Laplante [16] suggests to adopt formal methods for the verification of the specification of critical systems, and reducing as much as possible ambiguities.

In this context, a valuable contribution is in the work by Yue et al. [29] where an approach to generate a finite state machine from user requirements is provided. The state machine, expressed in an ad-hoc language, is derived from a modified version of a misuse case diagram. Nevertheless, work addresses exclusively systems functional requirements while we are interested also in non-functional ones. Gomes et al. [8] propose on a model-driven approach that allows deriving a system state machine from use case diagrams. The purpose of the approach is, in this case, to generate executable code of the system rather than the verification of the system specifications.

Jamhour and Penna proposed a study of the restoration (i.e., recovery) schemes for optical networks based on Continuous Time Markov Chains (CTMCs) [11].

Trivedi et al. deal with the modeling of non functional properties [18]: in this paper, vulnerabilities of an intrusion-tolerant system are evaluated considering both the attacker behavior and system response. In such a work, a state-transition model is proposed that takes into account each behavior of the system under attack. Starting from the full-working state of the system, the authors consider consider each phase of the attack (error detection, assessment of damage, recovery and the treatment of the attack) and produce a finite-state machine representing global levels of service of the system. The approach of Knight et al. [15] addresses the survivability of critical systems: in this work, the main difference between a reliable and a survivable system is that the former has to guarantee the same level of service, whereas the latter has to guarantee the essential services when failures happen. For these reasons the concept of graceful degradation *is introduced. The running example of the paper, in fact, shows a state machine describing a set of systems service modes and possible transition between them.*

7. Conclusion

Security is a primary issue in the design of modern high interconnected critical infrastructures. In this paper, model-driven techniques have been proposed to automatically obtain a survivability assessment, which that can be used for performing formal verification as well as in peer-review with system stakeholders. The approach exploits model transformations for the generation of the SAM from the MUCD. In particular, the model-to-model transformation is able to infer from the MUCD single-event chains triggering the transitions from one service mode to another. In addition, model queries are used to verify some simple properties on the generated SAM with the aim of providing feedback to the engineer about missing survivability strategies.

As shown in the paper, the transitions in the survivability assessment model represent exclusively single attacks or single repair actions. This can constitute a limitation of the approach that we plan to overcome in future works. Nevertheless, the approach constitutes a first step in the generation of a complete SAM from the UML models. Future work will extend the results of this paper in order to provide support for performing a more fine grained analysis: in particular, quantitative evaluation of the likelihood of attacks success will be investigated.

Acknowledgments

S. Bernardi and J. Merseguer have been supported by the European Commission under the H2020 Research and Innovation Action [DICE, Grant Agreement No. 644869], the Spanish Ministry of Economy and Competitiveness [ref. CyCriSec-TIN2014-58457-R], and the Aragonese Government [ref. T94, DIStributed COmputation (DISCO)].

References

[1] Agence France-Presse. Massive power failure plunges 80% of Pakistan into darkness. *The Guardian*, January 2015.

[2] E. Ancillotti, R. Bruno, and M. Conti. The role of communication systems in smart grids: Architectures, technical solutions and research challenges. *Computer Communications*, 2013.

[3] S. Bernardi, L. Dranca, and J. Merseguer. A model-driven approach to survivability requirement assessment for critical systems. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, 230(5):485–501, 2016.

[4] A. Cockburn. *Writing Effective Use Cases.* Addison Wesley, 2001. ISBN: 9780321605801.

[5] L. Coppolino, S. D'Antonio, and L. Romano. Exposing vulnerabilities in electric power grids: An experimental approach. *International Journal of Critical Infrastructure Protection*, 7(1):51–60, 2014.

[6] R. J Ellison, R. C. Linger, T. Longstaff, and N. R. Mead. Survivable network system analysis: a case study. *IEEE software*, 16(4):70–77, 1999. DOI:10.1109/52.776952.

[7] U. Gentile, S. Marrone, N. Mazzocca, and R. Nardone. Cost-energy modelling and profiling of smart domestic grids. *IJGUC*, 7(4):257–271, 2016.

[8] L. Gomes and A. Costa. From use cases to system implementation: statechart based co-design. In *Formal Methods and Models for Co-Design, 2003. MEMOCODE '03. Proceedings. First ACM and IEEE International Conference on*, pages 24–33, June 2003.

[9] C. Helman. Rolling Blackouts Force Texas To Import Power From Mexico. *Forbes*, March 2011.

[10] I. Jacobson, M. Christenson, P. Jonsson, and G. Overgaard. *Object Oriented Software Engineering: A Use Case Driven Approach.* Addison Wesley, 1992. ISBN:9780201544350.

[11] E. Jamhour and M.C. Penna. A reversible CTMC model for availability analysis of shared mesh restoration schemes for WDM networks. *Journal of High Speed Networks*, 20(4):223–237, 2014.

[12] C. W. Johnson. Analysing the Causes of the Italian and Swiss Blackout, 28th September 2003. In *Proceedings of the 12th Australian Workshop on Safety Critical Systems and Software and Safety-related Programmable Systems*, pages 21–30, 2007. ISBN:9781920682675.

[13] F. Jouault and I. Kurtev. Transforming models with atl. In *Proceedings of the 2005 International Conference on Satellite Events at the MoDELS*, pages 128–138, 2006. DOI:10.1007/11663430_14.

[14] S.M. Kaplan. Smart grid. electrical power transmission: Background and policy issues. *The Capital.Net, Government Series*, pages 1–42, 2009.

[15] J.C. Knight, E.A. Strunk, and K.J. Sullivan. Towards a rigorous definition of information system survivability. In *DARPA Information Survivability Conference and Exposition, 2003. Proceedings*, pages 78–89 vol.1, April 2003. DOI:10.1109/DISCEX.2003.1194874.

[16] P. A. Laplante. *Requirements engineering for software and systems.* CRC Press, 2013. ISBN:9781466560826.

[17] R.H. Lasseter. Smart distribution: Coupled microgrids. *Proceedings of the IEEE*, 99(6):1074–1082, June 2011. DOI:10.1109/JPROC.2011.2114630.

[18] B.B. Madan, K. Gogeva-Popstojanova, K. Vaidyanathan, and K.S. Trivedi. Modeling and quantification of security attributes of software systems. In *Proc. of the International Conf. on Dependable Systems and Networks (DSN 2002)*, pages 505–514, 2002.

[19] S. Marrone and U. Gentile. Finding resilient and energy-saving control strategies in smart homes. *Procedia Computer Science*, 83:976 – 981, 2016. DOI:10.1016/j.procs.2016.04.195.

[20] S. McLaughlin, P. McDaniel, and W. Aiello. Protecting consumer privacy from electric load monitoring. *Proceedings of the ACM Conference on Computer and Communications Security*, pages 87–98, 2011.

[21] T. Sato, D.M. Kammen, M. Macuha, B. Duan, Z. Zhou, M. Tariq, J. Wu, and S.A. Asfaw. *Smart Grid Standards: Specifications, Requirements, and Technologies.* John Wiley & Sons, 2015. ISBN:9781118653791.

[22] G. Sindre and A.L. Opdahl. Eliciting security requirements with misuse cases. *Requirements Engineering*, 10(1):34–44, 2005. cited By 382.

[23] G. Sindre and A.L. Opdahl. Eliciting security requirements with misuse cases. *Requir. Eng.*, 10(1):34–44, 2005. DOI:10.1007/s00766-004-0194-4.

[24] Symantec. *Severity Assessment – Threats, Events, Vulnerabilites, Risks*, (accessed January 2017). Available at: `https://www.symantec.com/content/en/us/about/media/securityintelligence/SSR-Severity-Assesment.pdf`.

[25] The Smart Grid Interoperability Panel Cyber Security Working Group. *Introduction to NISTIR 7628 Guidelines for Smart Grid Cyber Security*, September 2010. NISTIR Technical Report, available at: `https://www.nist.gov/document-13017` (accessed January 2017).

[26] US Department of Energy. *Benefits of Demand Response in Electricity Markets and Recommendations for Achieving them.* Available at: `https://emp.lbl.gov/sites/all/files/report-lbnl-1252d.pdf` (accessed January 2017).

[27] W. Wang and Z. Lu. Cyber security in the smart grid: Survey and challenges. *Computer Networks*, 57(5):1344–1371, 2013. DOI:10.1016/j.comnet.2012.12.017.

[28] Y. Yan, Y. Qian, H. Sharif, and D. Tipper. A survey on smart grid communication infrastructures: Motivations, requirements and challenges. *IEEE Communications Surveys Tutorials*, 15(1):5–20, 2013. DOI:10.1109/SURV.2012.021312.00034.

[29] T. Yue, S. Ali, and L. Briand. Automated Transition from Use Cases to UML State Machines to Support State-Based Testing. In *Modelling Foundations and Applications*, volume 6698 of *LNCS*, pages 115–131. Springer Berlin Heidelberg, 2011. ISBN:9783642214691.

[30] K. Zhou, C. Fu, and S. Yang. Big data driven smart energy management: From big data to big insights. *Renewable and Sustainable Energy Reviews*, 56:215–225, 2016.