A Case Study in Requirements for Survivable Systems

Robert J. Ellison
Richard C. Linger
Thomas Longstaff
Nancy R. Mead

Abstract

Increasing societal dependency on critical infrastructure systems is driving emergence of a new category of requirements engineering that addresses survivability objectives. This paper presents a case study in survivability requirements analysis. Survivability is the capability of a system to fulfill its mission, in a timely manner, in the presence of attacks, failures, or accidents. The Survivable Network Analysis (SNA) method permits assessment of survivability strategies at the requirements and architecture levels. Steps in the SNA method include mission requirements and architecture definition, essential capability definition, compromisable capability definition, and survivability analysis. Essential service scenarios and intrusion scenarios play key roles in the method. Survivability requirements must be defined for intrusion resistance, recognition, and recovery. This case study summarizes the application and results of applying the SNA method to a subsystem of a large-scale, distributed healthcare system. The study recommended specific modifications to requirements to support survivability objectives.

1 System Survivability Concepts

As part of its Survivable Network Systems Initiative, the CERT* Coordination Center of the Software Engineering Institute (SEI) is developing technology and methods for analyzing and designing survivable network systems [1], [2]. Survivability is defined as the capability of a system to fulfill its mission, in a timely manner, in the presence of attacks, failures, or accidents. Unlike traditional security measures that require central control and administration, survivability addresses highly distributed, unbounded network environments with no central control or unified security policy. Survivability focuses on delivery of essential services and preservation of essential assets, even when systems are penetrated and compromised. As an emerging discipline, survivability builds on existing disciplines, including security, fault tolerance, and reliability, and introduces new concepts and principles.

2 The Survivable Network Analysis Method

A primary focus of the SEI effort has been development of the Survivable Network Analysis (SNA) method for assessing and improving the survivability of network architectures, as depicted in Figure 1.

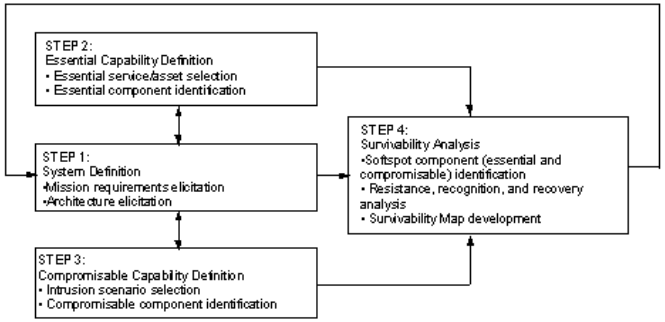

Figure 1: Steps in the Survivable Network Analysis Method

The method can be applied to an existing or proposed system by a small team of trained evaluators through a structured interaction with system personnel of several days duration. The method is composed of four principal steps, as follows:

Step 1. The mission requirements and architecture of the current or candidate system are elicited from stakeholders and system architects. Mission requirements deal with the overarching goals and objectives that the system must satisfy. These requirements are typically elaborated into specific functional and non-functional requirements for system services.

Step 2. *Essential services* (services that must be maintained during attack) *and essential assets* (assets whose integrity, confidentiality, availability, and other properties must be maintained during attack) are identified, based on mission objectives and consequences of failure. These services and asset uses are characterized by *usage scenarios* that are mapped onto the architecture to identify corresponding *essential components* (components that must be available to deliver essential services and maintain essential assets).

Step 3. *Intrusion scenarios* are selected based on the system environment and assessment of risks and intruder capabilities. These scenarios are likewise mapped onto the architecture to identify corresponding *compromisable components* (components that could be penetrated and damaged by intrusion).

Step 4. *Softspot components* of the architecture are identified as components that are both essential and compromisable, based the results of steps 2 and 3. The softspot components and the overarching requirements are then analyzed for three key survivability properties, namely, *resistance, recognition*, and *recovery.* Resistance is the capability to repel attacks. Recognition is the capability to detect attacks as they occur, and to evaluate the extent of damage and compromise. Recovery, a hallmark of survivability, is the capability to maintain essential services and assets during attack, limit the extent of damage, and restore full services following attack. The analysis of the "three R's" is summarized in a *survivability map,* as depicted in Figure 2. The map enumerates, for every intrusion scenario and corresponding softspot effects, the current and recommended *requirements strategies* for resistance, recognition, and recovery. The survivability map provides feedback to the original requirements, and may result in an iterative process of survivability evaluation and improvement.

Intrusion Scenario	Softspot Effects	Requirements Strategies for à	Resistance	Recognition	Recovery
(Scenario 1) …		Current			
		Recommended			
(Scenario n)		Current			
		Recommended			

Figure 2: Survivability Map Template at Requirements Level

3 Sentinel: The Case Study Subsystem

Management of mental health treatment is often performed as a manual process based on hand-written forms and informal communication. Substantial time and effort are consumed in coordination of various treatment providers, including physicians, social service agencies, and healthcare facilities. Carnegie Works, Inc. (CWI) is developing a large-scale, comprehensive management system to automate, systematize, and integrate multiple aspects of regional mental health care. The CWI system, named Vigilant, will ultimately be composed of some 22 subsystems operating on a distributed network of client and server computers, and will maintain a large and complex database of patient and provider records.

A vital part of the Vigilant system is development and management of *treatment plans*. A treatment plan is developed for a *patient* by a *provider*. The *problems* of each patient are identified, together with a set of *goals* and *actions,* including medication and therapy, to achieve those goals. Each treatment plan is carried out by an interdisciplinary and interorganizational *action team* composed of providers. An *affiliation* is an organization that provides healthcare services, possibly to many patients. Treatment plan development and management and action team definition and coordination are key functions of the Sentinel subsystem. As a subsystem of Vigilant, Sentinel interacts with providers, affiliations, and other subsystems. It maintains the action teams and treatment plans as part of the Vigilant patient database, and applies regulatory and business rules for treatment plan development and validation.

Because of the critical nature of mental health treatment, the need to conform to regulatory requirements, and the severe consequences of system failure, survivability of key Sentinel capabilities has been identified by CWI personnel as extremely important.

The process used in applying the SNA analysis method has been described more fully in another report [3]. In this paper we will focus on the requirements aspects of the case study.

4 Elicitation of Essential Service Scenarios

Requirements for large-scale systems typically specify a substantial number and variety of services and their usage scenarios for all classes and types of users. These services can exhibit substantial variations in properties such as frequency of use, time constraints, and criticality to mission objectives. Some services, for example, stock buy and sell orders, are invoked minute-by-minute, and must satisfy strict time constraints. The continuous availability and timeliness of such services are usually essential to mission objectives. Other services, for example, production of quarterly investment reports, are less frequently invoked, and their use can often be postponed if necessary due to adverse conditions.

Essential services are a subset of total system services, and themselves may exhibit varying degrees of essentiality. The starting point for definition of essential services is typically a set of usage scenarios that characterizes the normal use of the system.

Normal usage scenarios

Normal usage scenarios (NUS) are often developed during the requirements process in order to validate the requirements. In this case, the normal usage scenarios had been developed and were included in the Sentinel requirements documentation. The following normal usage scenarios elicited from the requirements documentation characterize the principal mission objectives of the subsystem. Each description below includes a statement of the primary Sentinel responsibility with respect to the scenario:

- NUS1: Enter a new treatment plan. A provider assigned to a patient admitted into an affiliation performs an initial assessment and defines a treatment plan, specifying problems, goals, and actions. Sentinel must apply business rules to treatment plan definition and validation.
- NUS2: Update a treatment plan. A provider reviews a treatment plan, possibly adding or changing problems, goals, or actions, and possibly updating the status of these items. Sentinel must apply business rules to treatment plan update and validation.
- NUS3: View a treatment plan. A provider treating a patient views a treatment plan to learn the status of problems, goals, and actions. Sentinel must ensure that the plan displayed is current and valid.
- NUS4: Create or modify an action team. A provider defines or changes the membership of a treatment team in an affiliation for a patient. Sentinel must ensure that the treatment team definition is current and correct.
- NUS5: Report the current treatment plans in an affiliation. An administrator views the current state of her affiliation's treatment of a patient or set of patients. Sentinel must ensure that the treatment plan summaries are current and correct.

- NUS6: Change patient medication. A provider changes the medication protocol in a treatment plan for a patient, possibly in response to unforeseen complications or side effects. Sentinel must ensure that the treatment plan is current and valid.

Essential Service Scenarios

Once the normal usage scenarios were understood, we were able to work with the client to identify the essential services and assets and their associated scenarios. This elicitation process took place in a work session, where we reviewed the normal usage scenarios, and discussed the way in which the system would be used in order to try to identify essential services and assets.

Essential services and assets represent critical system capabilities that must survive and be available during intrusions. Criticality is based on analysis of mission objectives, risks and consequences of failure, and availability of alternatives. Such an analysis may result in selection of any number of essential services and assets, and may stratify them into survivability classes of varying criticality.

The survivability analysis of the Sentinel subsystem was carried out together with CWI personnel, and was based on the normal usage scenarios identified in step 1 of the SNA method. The analysis resulted in selection of a single essential service, namely, NUS3, the capability to view treatment plans. This service, more than any other, was deemed essential to delivery of mental health treatment because providers depend on real-time, on-demand access to treatment plans in clinical situations, particularly in cases of medication or therapeutic problems of an emergency or life-critical nature. The other normal usage scenarios could be postponed for hours or even days in the event of system intrusion and compromise. The analysis also identified a single essential asset, namely, the treatment plans themselves. Preservation of treatment plan integrity and confidentiality was deemed essential to meeting Sentinel mission objectives. The other Sentinel artifacts, such as action teams, affiliations, and providers, could all be reconstructed or updated hours or days after intrusion with no irreversible consequences.

5 Requirements Affected by Essential Service Scenarios

Of course, there are many requirements that must be specified in order to support the usage scenarios. In this section, we give an overview of general Sentinel requirements and discuss in detail only those requirements that are affected by survivability considerations.

Functional requirements

The broad categories of functional requirements for the Sentinel subsystem are as follows:

- Create a treatment plan
- Check out a treatment plan
- Check in a treatment plan
- Modify a treatment plan
- Abandon changes
- Create an action team
- Assign an action team
- Change the composition of an action team
- Add coordinator
- Change coordinator
- View a treatment team
- View plan

In the original Sentinel requirements, treatment plans can undergo revision while incomplete, but must be validated when the plan is considered complete. Validation requires that a set of rules is checked. The rules are as follows:

- Each action may only be provided by one action team.
- Every action must be provided by a non-empty action team.
- There is exactly one coordinator for the patient.
- The coordinator must be a provider.
- Non-providers should take the guest role
- Persons having the guest role should be non-providers

This validation is performed when a treatment plan is ‘checked-in’. A treatment plan may be checked in if it is already checked out, and it is a valid treatment plan according to the set of rules.

Non-functional requirements

Non-functional requirements are specified in the areas of maintainability,

extensibility, scalability and system distribution. In addition, there are implied availability requirements and there is an operational environment that also can specify system requirements.

Availability requirements

Availability requirements are also very important in this system. There is a set of requirements that have to do with viewing the data in the database on demand. Specifically, there is a requirement to view treatment plans on demand. There is also a requirement to view treatment plan history, but this is not considered to be an essential service for survivability purposes.

Maintainability requirements

In the initial implementation of the system, there was no security requirement, however, there was a maintainability requirement to allow new features, such as security, to be easily added to the system. Note the seeming contradiction in the fact that the system does not have a security requirement, but it does have requirements for availability and for valid data, which may not be met in the event of an intrusion.

Operational environment

There is a modest level of security provided in the operating environment provided by login and password checking. There may also be some security built into the commercial database that is selected, but none of this discussion appears in the requirements. In addition, periodic backups were required, but were specified to be performed only on a weekly basis.

6 Development of Intrusion Scenarios

To begin the process of modeling potential intrusion activity, the requirements were studied to determine potential *motive* that an intruder might have in using the proposed Sentinel system. Experience with Internet intrusions indicate that there are several categories of attackers that potentially would have interest in attacking the Sentinel system. An analysis of security incidents provided the following list of attackers [4]:

- Hackers
- Spies
- Terrorists
- Corporate Raiders
- Professional Criminals
- Vandals

Within the Sentinel system, the relevant categories were hackers, corporate raiders, professional criminals, and vandals. It was determined that this system would not contain data that would politically motivate spies or terrorists. For the other categories, the following motivations were considered:

Hackers	• Curious about medical records (especially on celebrities and public figures) • Access as part of a larger sweep of networks regardless of application • Badge of merit to access medical records
Corporate Raiders	• Change of patient records to help a particular provider succeed or discredit another provider • Control the resources provided to patients • Change doctor recommendations to cut costs

Professional Criminals	• Manipulate providers and patient care to commit fraud
Vandals	• Destroy parts of the system to prevent access • Maliciously modify records to hurt patients • Make random changes

For each of these motivations, access to the Sentinel system is required. To find access routes that are feasible, it is important to hypothesize what access each group is likely to have to the Sentinel system.

Intrusion Scenario Selection

Based on the system environment and assessment of intruder objectives and capabilities, the following five intrusion usage scenarios (IUS) were selected as representative of the types of attacks to which Sentinel could be subjected. Each scenario is preceded by an IUS number and type of attack (shown in parentheses), and followed by a brief explanation:

- IUS1 (Data Integrity and Spoofing Attack): An intruder swaps the patient identification of two validated treatment plans.

  Sentinel performs validation of treatment plans before entering them into the database. In this scenario, an intruder accesses the database server to corrupt treatment plans without using the Sentinel client, but rather by spoofing a legitimate client.

- IUS2 (Data Integrity and Insider Attack): An insider uses other legitimate database clients to modify or view treatment plans controlled by Sentinel.

  The database security assumes that clients have exclusive write access to specific database tables. While the IUS1 scenario attempts to access the database directly, this scenario examines inappropriate access through other database clients.

- IUS3 (Spoofing Attack): An unauthorized user employs Sentinel to modify or view treatment plans by spoofing a legitimate user.

  Some terminal access points for Sentinel are located in public areas, and hence are not as physically secure as those in private offices. This scenario illustrates opportunistic use of an unoccupied but logged-in terminal by an illegitimate user who spoofs the legitimate logged-in user.

- IUS4 (Data Integrity and Recovery Attack): An intruder corrupts major portions of the database, leading to loss of trust in validated treatment plans.

  Scenarios IUS1 and IUS2 assume a sophisticated attacker who targets and recognizes specific treatment plans, and modifies only a few fields. This scenario assumes a brute-force corruption of the database, leading to large-scale loss of trust and potential denial of service during massive recovery operations.

- IUS5 (Insider and Availability Attack): Intruder destroys or limits access to the Sentinel software so it cannot be used to retrieve treatment plans.

  This scenario could be as simple as removing the Sentinel software, or could involve attacks on the network or application ports to limit application access.

Once the intrusion scenarios were selected, we were able to map each intrusion scenario to the requirements, and identify the current strategies for resistance,

recognition, and recovery. We could then make recommended requirements changes to enhance the "three R's".

7 Recommended Requirements Changes

In this section we discuss recommended requirements changes for the purpose of improving survivability of essential services and assets.

Functional requirements

The original requirements support validation of treatment plans on check-in. Data integrity can be improved by also validating treatment plans when they are checked out. In addition, validity of treatment plans can be checked continuously as an 'idle time' activity. An encrypted checksum field can be added to each treatment plan record, and validated during check-in, check-out, and during continuous validation.

Example requirements are:

- Treatment plans shall be validated when they are read and written. If a treatment plan is invalid, the last valid version of the treatment plan shall be recovered.
- Encrypted checksums shall be used to protect the integrity of the treatment plans.

Availability and integrity requirements

In order to allow treatment plans to be viewed after an intrusion, a requirement for an emergency reporting system can be added. In addition, integrity can be improved by selecting a commercial database that supports replication.

- An emergency reporting system shall allow treatment plans to be viewed during recovery.
- The selected database software shall support replication.

Operational environment

Modest improvements in the operational environment can enhance availability and the ability to recover:

- The Sentinel software shall be backed up on CD.
- Daily backups of the database shall be performed.
- Workstations located in public areas shall have a short timeout based on inactivity. There shall be login access thresholds for incorrect logins.

8 Conclusions and Issues

We were able to execute the four process steps in our model: elicitation, essential capability definition, compromisable capability definition, and survivability analysis. We felt that we succeeded in identifying the appropriate softspots and making recommendations for requirements and architecture changes for the purpose of improved survivability. Our analysis also resulted in reflection on some of the implied system requirements, such as availability.

The Sentinel case study showed that with relatively modest changes in requirements, a significant improvement could be made to the ability of the system to survive an intrusion. In addition, the process helped the client to focus on and articulate the essential services and assets of the system. When we first discussed the intrusion scenarios, it appeared that to the client that the system was very vulnerable, and that it might be very costly to improve survivability. Indeed, there is a cost/benefit tradeoff that must be considered depending on the desired level of survivability. The proposed changes in requirements, however, made it clear that improving system survivability does not need to incur an excessive cost. As in most software problems, it is much easier to improve survivability by examining the proposed system in the requirements and architecture phases, than it is to try to patch it later. Unfortunately, most of today's systems add security considerations in the form of operational patches. The fact that we did not have to recommend extensive changes was also a reflection of a good system design that made it easy to accommodate our

recommendations.

An outstanding issue has to do with expression of requirements. In some respects survivability requirements are like reliability requirements. It is not possible to ensure 100% survivability of a system, but it is possible to significantly raise the probability that essential services and assets will survive an intrusion. More research is required in survivability modeling, to develop strong methods for expression and analysis of survivability requirements. Today we have very good reliability models that can be used to support reliability requirements [5]. We need the same sort of robust modeling techniques in the survivability area.

9 Future Plans

This paper documents a case study in survivable systems requirements analysis. We believe that more such studies are needed to refine and validate the SNA method. For example, in this case, there was good synergy between the client and the development team. That is not always the case. The client was able to make decisions about requirements without having to consult others. In some commercial situations, for example, several groups participate in requirements development so that the decision process involves more than just one person or group.

In addition, we intend to explore techniques to better correlate survivability requirements with architecture components and their composite behavior. For this purpose, the idea of an architecture calculus may permit more complete and rigorous derivation of architecture behavior to determine if survivability requirements are satisfied. We are also investigating requirements definition for emergent algorithms, based on the idea that survivability is an emergent property of large-scale networks, and not necessarily present in individual nodes.

References

1 *Survivable Network Systems: An Emerging Discipline*, Technical Report CMU/SEI-97-TR-013, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, November 1997.

2 Linger, R. C., Mead, N. R., and Lipson, H. F., *Requirements Definition for Survivable Network Systems,* International Conference on Requirements Engineering, Colorado Springs, CO, IEEE Computer Society, Available online at http://www.cert.org/research, 1998.

3 Ellison, R.J., Linger, R.C., Longstaff, T., and Mead, N.R., *A Case Study in Survivable Network System Analysis*, CMU/SEI-98-TR-014, Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA, September 1998.

4 Howard, J., *An Analysis of Security Incidents on the Internet, 1989-1995*, Doctoral Dissertation, Carnegie Mellon University, Pittsburgh, PA, April 1997.

5 Musa, J.D.; Iannino, A.; & Okumoto, K. *Software Reliability: Measurement, Prediction, and Application.* New York, NY: McGraw-Hill, 1987.