

Survivability Metrics—A View from the Trenches¹

Partha Pal, Richard Schantz, Franklin Webber
BBN Technologies
10 Moulton Street, Cambridge, MA 02138
{ppal, schantz, fwebber}@bbn.com

Abstract

In this paper we describe our latest experience in evaluating a survivable system. This effort signified an unprecedented attempt to specify quantitative survivability metrics and to evaluate the system against them. Even though one way to quantitatively score the survivable system was demonstrated- a significant step forward in the context of survivability validation in its own right- it was also apparent that such quantitative measurements, by themselves, did not adequately establish the assurance case for the survivable system, and more research is needed in this area. With the advantage of 20-20 hindsight, we outline a number of thoughts about survivability metrics that are more amenable to assurance cases.

1. Introduction

From late 2002 to early 2005, a BBN-led team (referred to as the Blue Team henceforth) designed a survivability architecture, used it to defense-enable an undefended system and subjected the resulting system to multiple, largely unrestricted Red Team evaluations. This was part of the DARPA OASIS Dem/Val program, which sought to demonstrate that a new high-water mark in intrusion-tolerance and survivability is achievable using currently available technologies as building blocks. It intended to also demonstrate the higher level of survivability in an assured and quantifiable way.

Red Team exercises are often used and are currently the gold standard for evaluating cyber-defense. Previous (circa 2001-2002) DARPA Red Team exercises [1] have shown survival time only on the order of 20 minutes in experiments that focused on specific defense mechanisms or applications and included various rules of engagement restricting the

attacker. The OASIS Dem/Val (ODV) survivability vision involved experimenting with a DoD relevant information system in a nearly unconstrained manner.

To steer researchers towards quantifiably higher levels of survivability, DARPA prescribed a set of metrics. Considerable effort was spent on interpreting the metrics and devising ways to quantitatively evaluate the defense-enabled system against them. This was by far the most comprehensive attempt to deconstruct “survivability” into a number of measurable criteria against which a system can be scored.

However, the link between the scores obtained from experimental measurements and the assurance case that can be made about the defense-enabled system turned out to be somewhat weak. This experience exposed the need for more research, and provided a number of key insights about effective survivability metrics and survivability evaluation that are worth sharing.

2. The system under test

The Joint Battlespace Infosphere (JBI) [2] concept, developed by the US Air Force Research Laboratory (AFRL), seeks to establish effective interaction among disparate military computer systems that must exchange information in support of various network-centric warfare activities. The JBI aims to achieve this goal by using a publish-subscribe framework that allows diverse computing systems to interact in a decoupled manner as long as they follow a common API for defining Information Objects (IO) and for performing publish, subscribe and query (PSQ) operations. A JBI instantiation is therefore the core service facilitating the PSQ operations and a set of applications that are needed to execute a specific mission. One such instantiation, simulating the execution of an Air Tasking Order (ATO) and planning of a concurrent

¹ This research was funded by DARPA under AFRL contract No. F30602-02-C-0134.

airlift through the theater, was used as the demonstration vehicle in the OASIS Dem/Val program.

This exemplar JBI integrates applications for target nomination, monitoring environmental conditions, and creating ATOs. These applications are organized in 4 Local Area Networks (LANs, which are often referred to as enclaves) namely, the Planning LAN, the Environmental LAN, the Wing Operations LAN, and the AMC CONUS LAN after their respective Air Force functions. A representation of this system is shown in Figure 1. The core services are organized in their own LAN. A successful mission would involve making the go/no-go decision on an ATO that may have WMD sites as targets. The factors influencing the go/no-go decision include presence of WMD sites in the ATO as targets, the predicted weather condition in the target area, presence of friendly force near by, possibility of other air traffic (such as the airlift mission) in the theater, etc. An adversary may subvert the mission by forcing it to delay its decision or to make a wrong decision, or by stealing the target information.

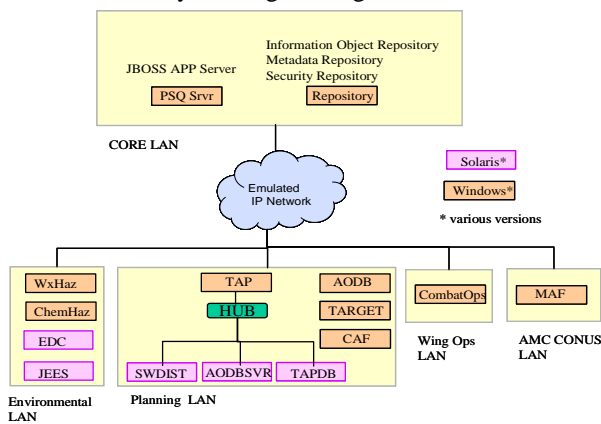


Figure 1: The Undefined JBI

The defense-enabled version of this system is known as the DPASA² survivable JBI. Description of the DPASA survivable JBI is not necessary for this paper; interested readers can find a description in [3].

3. Prescribed metrics

In an attempt to quantitatively set goals with multiple orders of magnitude improvement over the existing state-of-the-art, DARPA prescribed the following metrics for evaluating the survivable system:

1. 100% JBI critical functionality retained under sustained attack by a “Class A” Red Team with sufficient planning and preparation.

² DPASA is the project acronym and stands for Designing Protection and Adaptation into a Survivable Architecture.

2. 95% of large-scale attack detection within 10 minutes of attack initiation and 99% of attack detection within 4 hours of attack initiation with less than 1% false alarm rate.
3. Display meaningful attack state alarms.
4. 95% prevention of attacker objectives for 12 hours.
5. A factor of 1000 reduction of low-level alerts.

Several comments about this formulation are in order. First, the metrics reflect the evolution of “survivability” through multiple generations of security research. Instead of simply focusing on 3rd generation ideas like “tolerance”, it attempted to measure attributes relevant for earlier generations (e.g., “prevention” and “detection”) as well. Second, it also shows a conscious attempt to include lessons learned from past evaluations by incorporating false alarms and alert reduction; and by bringing potential sources of ambiguity and subjectivity to the fore by mentioning terms like “class A Red Team with sufficient planning and preparation”, “meaningful alarms”, and “within x hours of attack initiation”. Finally, it is worthwhile to note that these metrics, defined before the program inception, were early attempts to quantify survivability and reflect the state of the art for security validation at that time, as evidenced by multiple ways to capture the essence of “tolerance” (e.g., “retaining critical functionality” as well as “preventing attacker objectives”).

3.1. Blue Team’s interpretation

The Blue Team treated the prescribed metrics as design goals. Goals 1 and 4 were thought to be about *attack tolerance* and *survival*, and Goals 2, 3 and 5 about *intrusion detection* and *situational awareness*.

Attack tolerance and survival: The Blue Team thought that any functional operation (such as publication, receipt or delivery of an IO) performed within the mission period is “critical” if it is essential for the mission. The Blue Team also assumed that the schedule of such “critical” operations cannot be predicted, and in an adversarial situation not all operations will succeed all the time. Therefore, the Blue Team interpreted that as long as denial of service, corruption or disruption caused by the adversary is of short duration and does not affect the mission, “retention of critical functionality” goal will be met. This led to inclusion of a number of back up (redundancy) and recovery mechanisms in the system.

The Blue-Team assumed jeopardizing the mission is the ultimate objective of the adversary. This could be

achieved by successful compromise of any combination of confidentiality(C), integrity (I) and availability (A) of the PSQ services, or by other means. Toward that end, the adversary was expected to damage system components, and propagate attacks and unauthorized access throughout the system. Consequently, the survivability architecture attempted to make intrusion from outside or undetected movement from one part to another very difficult, in addition to defending the C, I, and A properties of data and executables,

Intrusion detection and awareness: Without getting into the details of what “large scale” meant, the Blue Team assumed the obvious interpretation of goal 2 that the system will have less time to detect (and react to) the attacks that are likely to cause widespread impact. This implied fortification of key paths and components with multiple overlapping Intrusion Detection Systems (IDS), policy enforcement and application embedded sensors. There was no precedent for designing IDS based on specified detection or false positive rates, so the Blue Team interpreted the other requirements to imply “best effort”.

3.2. Consensus interpretation

The Red Team evaluation was refereed by an independent White Team responsible for interpreting the metrics in a way that is agreed upon by the Red and Blue Teams, and also acceptable to external observers, technical experts and potential DoD consumers. The consensus interpretation that emerged prior to the actual exercise is summarized below.

Goal 1- Provide JBI critical functionality: Confidentiality (C), Integrity (I), and Availability (A) of PSQ operations were considered the JBI’s critical functions. The PSQ operations were divided into three classes for measuring this goal: 1) PSQ that critically supported a JBI mission, 2) PSQ performed by an artificially introduced scoring client (scorebot), and 3) PSQ that represented background traffic. Background PSQ was not used in scoring this goal.

Goal 2- Detect attacks: This goal was interpreted as a means for assessing the defended system’s ability to adequately reveal attacks launched against it. Detections were counted regardless of whether the attacks are successful or not. For scoring, this goal was divided into the following sub-goals:

1. **Goal 2a:** Detect 95% of large-scale attacks within 10 minutes of attack initiation.
2. **Goal 2b:** Detect 99% of attacks within 4 hours of attack initiation.
3. **Goal 2c:** Detect attacks with less than 1% false alarm rate.

Any perceived attacks reported by the Blue Team not attributable to Red Team activities were counted as false alarms. Attacks reported later than the stipulated time boundary were not considered detected.

Goal 3- Display meaningful alarms: An alarm was considered meaningful if it provided sufficiently accurate and timely information empowering the operator to engage human-in-the-loop responses and to stay aware of system’s automated responses.

Goal 4- Prevent attacks from achieving attacker objectives: This goal was used for evaluating the defended system’s ability to prevent the attacker objectives being met on a per-attack basis, where an “attack” was a sequence of Red Team actions intended to achieve an a-priori stated goal. If an attack achieved its objective, it was considered successful irrespective of the JBI mission outcome.

Goal 5- Reduce events: Events were interpreted to mean notices produced by detection mechanisms embedded in the defended system. Events were ranked and correlated to produce alarms that were the observable, situation-relevant information presented to the operators and security experts. The ratio of low-level events to operator alarms was the reduction factor measured for this goal.

3.3. Scoring rules

The White Team formulated a mechanism to score the defended system based on the consensus interpretation. It involved defining for each of the goals, a) one or more attributes that can be measured in the experiment, and b) scoring functions using these measurements to arrive at a quantifiable conclusion about the survivability criterion implied by that goal.

Scoring for goal 1: C, I, and A of critical operations were scored separately by counting the number of IOs for which these attributes were not violated during the exercise. Availability measurements depended on maximum allowable latencies determined by AFRL’s domain experts. Determination of integrity and confidentiality violations was based on evidence provided by the Red Team. In-transit corruption that was recovered before the consumption or corruption after consumption did not qualify as integrity violation. Availability score was defined as $((s/n)*100)$, where s is the number of PSQ operations completed within maximum allowable latencies. Integrity score was defined as $((u/n)*100)$, where u is the number of uncorrupted IOs. Confidentiality score was defined as $((c/n)*100)$, where c is the number of uncompromised IOs. In all of the above, n is the total number of non-background PSQ operations.

Scoring for goal 2: Whether an attack was large-scale was determined by independent evaluators, AFRL and the White team at the conclusion of the exercises. A large-scale attack was considered detected if the Blue Team reported it within 10 minutes of launch. For other attacks, the detection rate was defined as $((d/T)*100)$, where each attack launched by the Red Team was included in T, and d included those that were reported by the Blue Team within 4 hours of launch. The false-alarm rate was defined as $((f/A)*100)$, where A was the total number of alarms, and f is the number of reported alarms reported that cannot be attributed to Red Team attacks.

Scoring for goal 3: Meaningful alarm rate was defined as $((m/A)*100)$, where m is the number of alarms reported by the blue team containing meaningful information, and T is total number of alarms. Determination of what constitutes meaningful information was made at the end of the exercise by independent evaluators, AFRL, and the White team.

Scoring for goal 4: The prevention rate for scoring the survivable system against goal 4 was defined as $((p/T)*100)$, where p is the number of attacks that failed to produce the a-priori defined success indicators, and T is total number of attacks.

Scoring goal 5: The alert reduction factor required for scoring this goal was defined as (u/A) , where u is the number of unique events that contribute to an alarm, and A is the total number of alarms. Redundant devices or systems that use the same mechanism to detect and report the same event, and events that contribute to false alarms were not counted.

Table 1: Undefended system's scores

Goal 1 Criteria	Successful	Total	Percentage
Availability (PSQ)	53	69	76.81%
Availability (IO)	1722	1776	96.96%
Integrity (IO)	1722	1776	96.96%
Confidentiality (IO)	1695	1712	99.01%
Goal 2 Criteria	Detected	Total	Percentage
Large Scale Attacks	0	0	-
Attacks	4	17	23.53%
False Alarms	No Data	No Data	-
Goal 3 Criteria	Meaningful	Total	Percentage
Meaningful Alarms	No Data	No Data	-
Goal 4 Criteria	Prevented	Total	Percentage
Attacks	0	17	0%
Goal 5 Criteria	Events	Alarms	Reduction Ratio
Occurrences	No Data	No Data	-

3.4 Summary of scoring results

The results of scoring the undefended system under attack using the scoring functions are shown in Table 1. Results of scoring the defended system in two exercise

runs are shown in Table 2 and Table 3. These scores are obtained from the report produced by the White Team.

It is difficult to comprehend the Red Team's degree of success against the undefended system from Table 1. Despite the high Goal-1 scores, the Red Team was able to jeopardize the mission fairly quickly and easily. Valuable information (account IDs, passwords, and configuration data) were harvested from the system. As shown by the Goal-4 score, all attacks against the undefended system were successful. Combined with the high Goal-1 scores, it implies that jeopardizing the mission without significantly disrupting the PSQ operations was possible in the undefended system.

Table 2: Defense-enabled system's scores (exercise 1)

Goal 1 Criteria	Successful	Total	Percentage
Availability (PSQ)	86	120	71.67%
Availability (IO)	1994	2546	78.32%
Integrity (IO)	2060	2063	99.85%
Confidentiality (IO)	2060	2060	100%
Goal 2 Criteria	Detected	Total	Percentage
Large Scale Attacks	?	0	?
Attacks	5	9	55.56%
False Alarms	3	8	37.50%
Goal 3 Criteria	Meaningful	Total	Percentage
Meaningful Alarms	No Data	545	?
Goal 4 Criteria	Prevented	Total	Percentage
Attacks	6	9	66.67%
Goal 5 Criteria	Events	Alarms	Reduction Ratio
Occurrences	20663	545	37.91

In the first exercise, all attacks were availability attacks, 3 of which were partly successful in delaying publications. Table 2 shows that the availability scores of the defended system were lower than the undefended counterparts, implying a greater impact on PSQ operations. This is because unlike the undefended system, it was not possible to affect the mission without attacking the PSQ operations, so the Red Team had to go after availability of PSQ services. Another anomaly is the less than 100% integrity score when no attack on integrity was launched. Analysis showed that 3 retransmitted IOs that were previously received successfully were flagged as violating integrity. The asynchronous protocols of the survivable system sometimes retransmit even when the IO was received, and although the IOs in question never went in circulation, these three cases impacted the score. Goal-2 scores show that none of the 9 attacks were considered "large scale". Blue Team made 8 reports, 3 of which were deemed "false alarms", the remaining 5 were considered to have captured the Red Team attacks. For Goal-3, the White Team concluded that the Blue Team's reports and logs did not have enough information to determine meaningfulness. The

survivable system successfully deflected 6 of the 9 attacks as shown by Goal-4 score. For Goal-5, automated attack detection logs showed 20,663 low-level events, from which 545 alarms were raised, hence the 37.91 factor reduction.

Table 3: Defense-enabled system’s scores (exercise 2)

Goal 1 Criteria	Successful	Total	Percentage
Availability (PSQ)	1	31	3.23%
Availability (IO)	37	984	3.76%
Integrity (IO)	37	37	100%
Confidentiality (IO)	37	37	100%
Goal 2 Criteria	Detected	Total	Percentage
Large Scale Attacks	1	1	100%
Attacks	?	0	?
False Alarms	2	3	66.67%
Goal 3 Criteria	Meaningful	Total	Percentage
Meaningful Alarms	No Data	625	?
Goal 4 Criteria	Prevented	Total	Percentage
Attacks	0	1	0%
Goal 5 Criteria	Events	Alarms	Reduction Ratio
Occurrences	43049	625	68.88

In the second exercise, the Red Team launched one attack on the VPN routers guarding the various network enclaves of the defended system that took down all inter-enclave communication. This explains the low availability score and the 0 score of Goal-4. However, proper interpretation of the high integrity and confidentiality scores in Goal-1 needs to consider that only 37 IOs were published. This attack was classified as *large-scale*. After the Red Team began their attack, the Blue Team recognized that communications were disrupted on all of the client enclaves, and reported the attack within the 10-minute time window (hence the 100% score for Goal-2a). However, the Blue Team also made 2 misdiagnosed reports regarding the cause of the attack, hence the 2 false alarms. The White Team decided that there was not enough data to score Goal-3 in this run as well. For Goal-5, it was observed that the survivable system aggregated 43,049 low-level events to 625 alarms, achieving a reduction factor of 68.88.

3.5 Discussion

Despite considerable preparation, scoring involved a significant amount of subjectivity (e.g., was the alarm meaningful? was it a large scale attack? was it an attack step, or an attack in and of itself?). The scoring process demanded significant human discipline (e.g., proper journaling and record keeping by both the Red and Blue Teams) that tends to get ignored when the attack is at high tempo and the defense demands more attention. Some subjectivity and human discipline is perhaps inevitable in this context.

The list of attacks that were intrinsically prevented by the survivability architecture was compiled a-priori, with the idea that the system may not produce any alerts for these attacks. But, the Red Team also knew which attacks these were, and these attacks were never used. This implies that the scores did not consider the attacks that were prevented in that way.

The exercises showed that it is possible to score both the defended and undefended systems quantitatively against given survivability metrics. The metrics used were far more comprehensive than previously used. Although the scoring results showed that not all attributes were measurable equally well, and the scores were confusing at times, it was a non-trivial achievement to formalize and execute quantitative scoring of survivability in multiple live-fire exercises.

4. Additional considerations

Arguments for “comparative measures” focusing on the difference between undefended and defended systems have been made in the past. Despite the differences between the two systems and the attacks used against them, attacking both with the *same attacker objectives* is useful for assessing the incremental improvement. In OASIS Dem/Val both the defended and undefended systems were subjected to common attacker objectives (i.e., “steal an IO” or “jeopardize the mission”), and it was apparent that any such objective was much harder to achieve in the defended system. However, this was a qualitative observation. A high relative improvement from a low point might not have served DARPA’s original objective of achieving very high levels of survivability, but the fact that the scores did not adequately capture the improvement indicates the need for additional consideration.

Attacker work factor may be one candidate for such consideration. Attacker work factor can be conceived of as a combination of the preparation effort prior to actual attack, and the steps, complexity and sophistication of actions taken during the actual attack. In OASIS Dem/Val, the former did not matter because, to simulate a nation state adversary, the Red Teams were given the system design, code, and even time on the defended system for preparation. However, the second component can still be useful. A counter-argument against considering attacker steps/attack complexity is that the adversary could script the attack and run the entire attack as a single step. In that case, complexity of the script should be considered. The scripted attack may still involve multiple interactions with the system and take a measurable wall-clock time

to achieve its objectives. At the very least, the attack would need confirmation of the success (or failure) of its previous actions to mount the next (it is interesting to note that in OASIS Dem/Val one Red Team actually argued that the defended system did not provide enough visibility to the outcome of their actions, impeding their ability to mount multi-stage attacks). Furthermore, the defensive mechanisms may force off-script course-corrections. We need to find a way to use metrics like the attacker work factor in survivability evaluation without using it to judge the Red Team.

A single Red Team exercise is not enough to assess how many attacks were eliminated or how many defensive layers the Red Team had to overcome. Blue Team's internal evaluation of these issues showed that for each objective considered (from a reasonable but non-exhaustive sample), the adversary will need to overcome or bypass at least 2 defense mechanisms. Such data points are useful for the assurance case.

The two exercises using the defended system largely attacked the periphery and did not stress the internals of the survivability architecture. Therefore, DARPA initiated a 2nd phase of adversarial testing, where the Red Team worked with a Blue Team member to develop attacks that target the internals of the defended system, and placed the attack code inside the system unbeknownst to the Blue Team. About 20 of these short exercises were run to test very focused hypotheses like "can the mission be subverted if a PSQ server is compromised"? In 15 of such runs the mission was completed, but with considerable human participation. These runs exposed the pros and cons of the defense much more than what was achieved in earlier exercises.

5. Conclusion

Having experienced several attempts to evaluate survivability, we can outline a number of ideas that we think will be useful in future attempts. First, evaluation of survivability should not depend only on experimental techniques or scores obtained from experimental measurements. Quantitative measures can also be obtained from non-experimental techniques and they should also be used in constructing the assurance case. For example, the number of attacks or attack classes eliminated, or number of defense mechanisms that the adversary must overcome in order to achieve for each of a given set of attacker objectives can be estimated by logical analysis and white boarding. These numbers also provide very useful data points for rating an individual survivable system as well as comparing one with others.

Second, intervals, such as "time until certain event" seems to be very useful for quantifying survivability. The event in question could be a Red Team achievement, a scenario milestone, or the point at which the system degrades beyond being usable. A good experiment design should include at least one such "time to" metric, in addition to other applicable *time-based* (i.e., round-trip time, uptime, completion time of an operation) or *count-based* (i.e., number of alerts, number of successful or failed events) or *rate-based* (i.e., rate of successful transactions) metrics.

Third, easily measurable and intuitive metrics tend to be mission or application specific. However, making them too specific can be problematic. For instance, a successful publication in a redundancy-based PSQ protocol may involve multiple retries. In this case, counting low-level PSQ messages or IOs instead of the externally visible event of successful publication would be inappropriate. Counting preservation of C, I and A in IOs separately exposed some of these issues in OASIS Dem/Val. Very high C and I scores were possible with a low A score, seemingly indicating some success in "retaining critical functionality" despite a failure of the mission. Evaluating the effectiveness of specific defense-mechanisms is useful- a number of the 20 focused exercises in the 2nd phase were of this nature. However, we argue that high-level metrics and requirements should be formulated in terms of the system's external behavior.

Finally, while the value of multiple Red Team exercises is more than a single or a small number of exercises, it is also much more costly. The OASIS Dem/Val program demonstrated a way to conduct adversarial tests where the Red Team flavor can be retained in lightweight and focused exercises. This approach was quite useful in assessing the defended system and we believe represents another building block for making more comprehensive assurance cases.

6. Reference

- [1] Nelson, W., Farrel, W., Atighetchi, M., Clem, J., Shepard, M., and Theriault, K. "APOD Experiment 2: Final Report", *BBN Technologies LLC, Technical Memorandum 1326* (Sep. 2002).
- [2] AFRL JBI homepage: <http://www.infospherics.org>
- [3] Chong, J., Pal, P., Atighetchi, M., Rubel, P., and Webber, F. "Survivability Architecture of a Mission Critical System: The DPASA Example", *Proc. 21st Annual Computer Security Applications Conference* (Dec. 2005), 495-504.