BluGen: An Analytic Framework for Mission-Cyber Risk Assessment and Mitigation Recommendation

Thomas Llansó, Martha McNeil, Dallas Pearson, George Moore
The Johns Hopkins University Applied Physics Laboratory

Abstract

Systems security engineering (SSE) is a complex, manually intensive process, with implications for cost, time required, and repeatability/reproducibility. This paper describes BluGen, an analytic framework that generates risk plots and recommends prioritized mitigations for a target mission/system environment based on a stated level of threat and risk tolerance. The goal is to give working system security engineers a head start in their analysis. We describe BluGen in the context of Design Science Research and evaluate accordingly.

1. Introduction

As typically practiced today, systems security engineering (SSE) is a complex, manually intensive process, with implications for cost, time required, repeatability, and reproducibility [1][2]. A key driver in SSE is the need to develop an understanding of mission/business risk due to cyber attack (henceforth we simply say mission risk). An understanding of risk then informs mitigation prioritization. Risk-based approaches are increasingly mandated in government and industry with the arrival of standards such as the US National Institute of Standards and Technology (NIST) Risk Management Framework (RMF) [3] and the United States Department of Defense implementation of RMF [4].

Given the sometimes rapid evolution of mission, system, and threat, such assessments are perishable and must be periodically updated, with mitigations adjusted accordingly. Hence, the SSE community urgently needs ways to reduce cycle time and effort in the SSE process.

While many SSE-related automation tools exist, they tend to be narrowly focused in areas such as vulnerability scanning, e.g., Open VAS [5] and software static analysis, e.g., ESC/Java [6]; or they focus on capturing compliance/requirements data that subject matter experts (SMEs) manually create [7][8].

Thus, a gap currently exists for tools to help the security engineer work from threat through to security controls. Specifically, we have found that a gap exists for automation solutions that meet the requirements in Table 1, which we derived from performing SSE over a seven year period. The primary research question explored in this paper is whether creating an automated tool that meets the requirements in Table 1 and thus the gap identified, can provide SSEs with a measurable head start in their work, allowing more efficient use of their time to provide an engineering result at least as good as or better than one derived via manual methods.

Table 1 – Requirements

ID	Requirement
R1	Analyze threat, mitigation, and entity data to estimate attack exposure.
R2	Estimate attack consequence/impact based on criticality inputs (mission impact of compromises to data on a specific asset).
R3	Assess risk by generating risk plots of exposure vs. criticality based on R1 and R2.
R4	Recommend mitigations for entities based on risk assessed in R3 and a stated level of risk tolerance.
R5	Be able to analyze systems in the early concept phase as well as existing systems.
R6	Produce results that are repeatable and reproducible.
R7	Be usable in a range of environments, e.g., from enterprise IT settings to industrial control settings.
R8	In terms of time required to complete SSE, do no worse and ideally better than traditional manual SSE.
R9	Allow engineers to produce an SSE solution that is at least as good as one produced via manual SSE.

The research contribution of this paper is an entity- and capability-centric framework called BluGen that is intended to meet the requirements in Table 1.
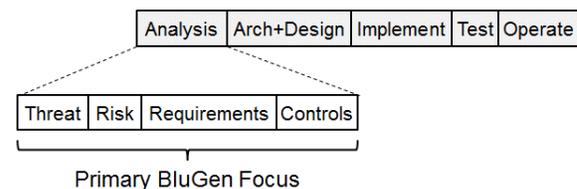

Figure 1 – Applicability of BluGen

The terms entity and capability are defined below. BluGen is primarily focused on SSE analysis steps from threat to security control identification that are part of the larger Systems Engineering (SE) process (Figure 1).

The theoretical foundation of BluGen rests on attack-centric risk assessment frameworks, as exemplified by NIST [9] and the International Standards Organization [10], where risk is a function of attack likelihood and mission consequence if the attack is successful. However, BluGen goes beyond attack-based risk approaches, as discussed in Section 2, by focusing on attacker/defender capabilities at the entity/asset level, with an exposure metric replacing likelihood of attack.

Figure 2 presents core BluGen constructs, with a small sampling of entity types. As with other figures in this paper, the figure is expressed using the Unified Modeling Language [11]. BluGen analyzes various entity types. The term "entity" refers to either missions supported by a target cyber environment, cyber assets, data processed, or the roles that people play in that environment. By "capability," we refer to (1) a particular proficiency, dubbed a "red capability," that an attacker can use to achieve a discrete step in a cyber attack or (2) a particular proficiency, dubbed a "blue capability," that can help to mitigate an attack. By "mitigate" we mean the capability to identify, prevent, detect, respond, and/or recover [12] from the effects of one or more red capabilities that are composed into a cyber attack. Entities are the targets of capabilities, both red and blue.
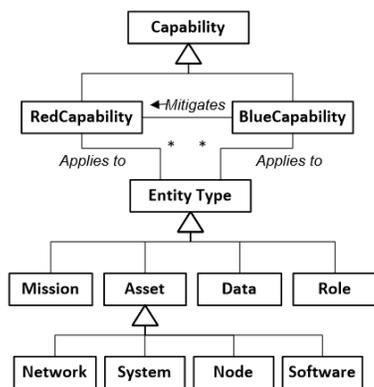

Figure 2 – Core constructs used in BluGen

An example of a red capability is "Uses moderately sophisticated social engineering techniques." An example of a blue capability that could help to mitigate this red capability is "Provides basic security-related training to information system users." The "entities" in this example are human users who play roles in the use/operation of a target information system.

The rest of this paper contains sections on background and related work, evaluation, artifacts, discussion, conclusions/future work, and references.

2. Background and Literature Review

In this section, we provide background and review of the literature and describe how SSE has often been superficially approached in the past, how SSE has been in need of stronger processes and related automation, and how vulnerability- and attack-centric approaches have thus far proven difficult to automate.

SSE is an SE sub-discipline that has as its primary goal the effective management of risk to mission objectives that could result from cyber-attacks against systems supporting those missions [13][14][9].

Historically, standards organizations and governments have put forward standards and policies that have not required strong SSE processes but have often instead focused on a compliance-oriented approach driven by checklists of security controls. For example, Federal Information Processing Standard 199 [15], CNSS Instruction 1253 [16], the DoD Information Assurance Certification and Accreditation Process (DIACAP), and [17] have unintentionally encouraged superficial security engineering analysis by not requiring substantive assessment of the cyber threat and risk that then informed prioritized mitigation to objectively bring the highest risks down to an acceptable level. In addition to government standards, industry standards, such as the Payment Card Industry Security Standards Council [18], have similarly encouraged a compliance-oriented approach to security.

Government and industry have begun taking the risks associated with the cyber threat more seriously in recent years [19]. For example, International Standards Organization (ISO) Standard 27001 [10] has a strong risk focus. In addition, the RMF now requires that risk assessments of the cyber threat be carried out as the basis for creating and prioritizing mitigation-related requirements [9][20].

So while the need to carry out risk assessments and select appropriate mitigations based on risk management decisions has now become widely acknowledged and written into broad-based policies, it is also increasingly recognized that commonly used risk assessment and mitigation engineering procedures tend to be excessively time consuming and potentially error-prone because they are fundamentally manual processes carried out by SMEs; automation, including related models and analytics, is thus required [21][22][23].

The degree to which SSE risk and mitigation processes can be automated depends in part on the underlying approach taken to risk assessment and SSE. Cyber assessment approaches have historically been vulnerability-centric, attack-centric, or some combination of the two.

2

In a vulnerability-centric assessment, e.g., [24] and [25], risk is lowered through a process of discovering and remediating (e.g., patching) vulnerabilities in the affected systems. Vulnerability scanning tools such as Nessus [26] and nmap [27] can assist in such assessments. Assessing risk based on identifying vulnerabilities, however, can at best identify only a small percentage of the total set of vulnerabilities in a system due to the difficulty of identifying so-called zero day vulnerabilities that are generally believed to be present in abundance in complex cyber-intensive systems [28]. Heart Bleed [29] and vulnerabilities present in Adobe Flash, e.g., [30], are two examples of serious, broad-based vulnerabilities that went undiscovered for years. A further difficulty in taking a vulnerability-centric approach is that systems that contain vulnerabilities may be frozen in their configurations, with patching disallowed because of the disruptive nature of doing so and/or the loss of "certification" of altering systems that have been approved for operation in a particular locked down configuration. Common examples of such systems include military and critical infrastructure systems.

Attack-centric approaches (e.g., [9][31][32]) analyze risk by enumerating potential cyber attacks and scoring each attack by likelihood of occurrence and mission impact. As with approaches that attempt to enumerate vulnerabilities, approaches that attempt to enumerate potential attacks are challenging because of the vast attack surfaces that complex cyber-intensive systems expose.

Attack-based methodologies typically require manual attack scoring that depends on cyber and mission SMEs for attack likelihood and impact scoring, respectively. However, manual scoring does not scale. For example, an attack-based risk analysis of a modest system (e.g., 4 mission threads, 40 nodes, 4 attack vectors, 3 attack effects, and 4 data items/node on average) can require an upper bound of 7,680 ($=4\cdot40\cdot4\cdot3\cdot4$) unique attack contexts to be scored for likelihood and impact.

Consequently, SMEs tend to consider a fraction of the attack surface using small, typically non-random samples with attendant concerns about how well such samples generalize to the entire attack surface. In addition, such assessments are time consuming and subject to the effects of SME-bias in assigning scores along ordinal scales. While some progress has been made in automating impact scoring, e.g., [33] and [34], approaches to automating attack likelihood scoring remain in their infancy. Furthermore, there is thus far no clear-cut automation path that leads from attack-centric risk assessment to mitigation analysis.

3. Method

We characterize and analyze BluGen in terms of Design Science Research principles [35][36]. In particular, we use Hevner's Information Systems Research Framework (Figure 3) to analyze BluGen, as discussed in the validation section later.
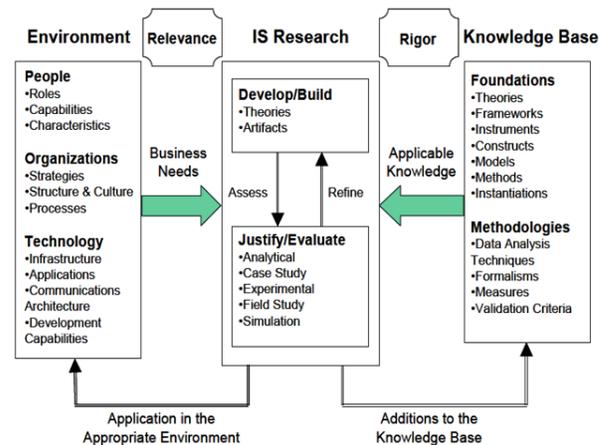

Figure 3 – Hevner IS Research Framework

We use the framework to evaluate the BluGen framework itself, along with two models and four methods contained within the framework and an instantiation of the framework.

4. Artifact Description

This section describes BluGen artifacts, which are summarized in Table 2.

Table 2 – Artifact Summary

Artifact Type	BluGen Artifact
Framework	BluGen itself
Models	Environment, Reference Catalog
Methods	Risk, Exposure, Criticality, Mitigation
Instantiation	BluGen proof of concept

Following a discussion of the design search process that we followed to arrive at the artifacts in the table, each of the BluGen artifacts is described.

Design Search

In terms of design as a search process for BluGen [37], we note that SSEs using attack-centric approaches like Mission Information Risk Analysis (MIRA) [38] tend to use some variant of the following basic steps when carrying out risk assessment and mitigation processes manually. The presumption in these steps is that the analyst will reference the Defense Science Board [19] threat tiers to conduct attack level of effort scoring. The tiers range from I to VI, with lower tier threat actors less capable than higher tier actors.

1. Identify a set of potential hypothetical attacks to study against the target system environment. For each attack, follow the steps below.

2. Determine the worst case mission impact if the attack is successful.

3. Break the attack into the most likely steps and determine the attack (red) capabilities needed in each step of the attack.

4. Set the overall attack level of effort/capability score equal to the highest Defense Science Board (DSB) tier associated with any of the red capabilities from step 3.

5. For attacks with impact and effort scores that exceed the risk tolerance levels of mission/system stakeholders, consider mitigation possibilities for each red capability in the attack, as described in the steps below.

6. For each red capability, determine the blue capabilities needed to mitigate the red capability.

7. Compare the blue capabilities already applied to the entity or entities under consideration.

8. If a needed blue capability is not already specified, then recommend that it be included.

We observed that the data and mappings associated with step 3, 6, 7, and 8 are invariant of the details of any given target environment, assuming the entity types in that environment are already known and mapped. This observation led us to the idea that capturing such data and mappings in a "Reference Catalog" (Figure 4) and reusing the data across SSE analyses could save engineers time and lead to more consistent scoring. Taking this idea a step further, we believed that cross referencing this data (e.g., threats and mitigations to entities and mitigations to threats) could be automated through an analytic that mimics the steps described earlier.
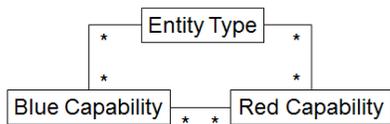

Figure 4 – Reference Catalog concept

The original concept behind Figure 4 was sketched in [39]. As the figure shows, blue capabilities that mitigate red capabilities are mapped to those red capabilities in a many-to-many fashion. Similarly, blue and red capabilities map many-to-many to entity types based on relevance, as not every capability is applicable to every entity type.

In relating capabilities to individual cyber attacks, we observe that attack capabilities possessed by a threat actor can be composed in various combinations and sequences to represent any possible cyber attack; that is, capabilities are the "atoms" from which attacks are constructed. We thus justify the capability-based approach on the basis that if we are able to use blue capabilities to mitigate (e.g., prevent or ameliorate the effect) at the red capability level, then any attacks composed from those red attack capabilities would be thus disrupted.

Figure 5 illustrates the concept. The figure shows five sample cyber attacks in the abstract broken down into individual steps. For this example, each attack uses "red capability 123" in one of its steps. Capability 123 might be, for example, "Can use brute force searches to defeat strong hashes protected by strong passwords." Preventing or disrupting the success of this particular capability would potentially prevent or disrupt attacks composed in part from this capability.

Figure 5 – Disrupting attack and attack effects by disrupting component capabilities used in attack steps

As red capability mitigation coverage becomes more complete, fewer attacks composed from those capabilities have a chance of succeeding because fewer unmitigated red capabilities are available from which to compose the attacks.

Models

BluGen has two main models: an Environment model and a Reference Catalog model.

Environment Model

The Environment model describes the target mission/cyber environment, any existing mitigations, and related analysis parameters that BluGen analyzes for risk and mitigations. The Environment model consists of the following elements: threat level expected, risk tolerance, environment description, and criticality, as discussed below.

The threat level expected is an integer in the range of one to six and represents a mapping to one of the threat tiers defined in Gosler [19]. Risk tolerance consists of two metrics called exposure and criticality that are expressed as percentages (range 0.0 to 1.0). The metrics are described below. Whenever an entity's exposure and criticality scores both exceed the corresponding risk tolerance values, BluGen recommends

mitigations to bring entity exposure down below the specified criticality level.

The environment description describes the target cyber environment to be assessed by BluGen and contains three key sets: M, a set of missions; E, a set of entities; and D, a set of data types. The entities in E support the missions in M by processing data in D. Below we follow the convention that variables i, j, and k index objects from E, M, and D respectively, with

- $e_i \in E$, $1 \le i \le |E|$;
- $m_j \in M$, $1 \le j \le |M|$; and
- $d_k \in D$, $1 \le k \le |D|$.

Each entity instance consists of a name, an optional description, an entity type, and a set of blue capabilities that have already been mapped to the entity. The entity type must map onto one of the entity types found in the Reference Catalog model (discussed below). If a new entity type is encountered that is not in the Reference Catalog, it must be added and mapped accordingly. One type of entity is the mission entity. For mission entities, the environment description includes the overall weight of each mission relative to the other directly supported missions; weights are typically determined by SMEs. Mission weights should sum to 1.0 for a given Environment model instance.

The criticality component of the Environment model consists of a set of so-called "raw" criticality 4-tuples. Each criticality triple, (m_j, e_i, d_k), is a unique combination of three values: a given mission, m, a given entity, e, and a given mission data element, d. Note that not every possible triple in the Cartesian product of M·E·D represents a viable combination, as not every data type is associated every entity, and not every entity is associated with every mission.

Associated with each raw criticality triple is a criticality score expressed as a percentage (range 0.0 to 1.0), with 0.0 meaning not mission-critical at all and 1.0 meaning maximal mission criticality. Criticality means the worst case mission impact ("mission kill") if a cyber compromise were to occur in the context defined by the triple. For example, one of many criticality triples for a robot might be: (m=navigate, e=sensor, d=location) and the worst case impact for the triple might be found to be 1.0 (due to, say, an integrity attack on d=location in the m=navigate and e=sensor context).

BluGen does not prescribe how raw criticality scores are derived; the scores could be assigned by SMEs or they could come about from running a mission performance model that can model cyber effects and automatically determine related mission impacts, e.g., [34]. The former would typically provide scores

along an ordinal scale, while the latter would typically provide scores along a ratio scale based on mission performance metrics. The latter is more desirable to avoid potential SME bias.

Reference Catalog Model

The other BluGen model is the Reference Catalog, which consists of (1) a set of red capabilities that attackers can compose into attacks, (2) a set of blue capabilities representing potential mitigations to red capabilities, (3) an entity type taxonomy, (4) a set of mappings between blue and red capabilities, and (5) a set of mappings between capabilities and entity types in the taxonomy, as given in Figure 4. Mappings of both types are many-to-many. A mapping of a blue capability to a red capability means that the blue capability can, potentially in concert with other blue capabilities, help to mitigate the corresponding red capability. A mapping from red and blue capabilities to a given asset type in the asset type taxonomy means that the capability is relevant to (can affect) the corresponding asset type.

The red capability set can be taken from a threat model, such as the DoD CIO/AT&L threat model [40]. The blue capability set can, in part, be derived from a security control catalog, such as NIST 800-53 [41]. We expect the entity taxonomy to be organically grown over time, though data mining against existing sources may be done. For example, the National Vulnerability Database can provide an initial set of asset types, where assets are one type of entity.

Methods

The BluGen framework supports four basic methods: Risk, Exposure, Criticality, and Mitigation.

Risk. The Risk (1) method computes mission risk due to cyber effects (e.g., attacks) for each entity, e_i, and is the product of two additional methods, Exposure and Criticality, described below.

$$Risk(e_i) = Exposure(e_i) \times Criticality(e_i) \qquad (1)$$

Exposure. The Exposure method computes how "exposed" (open to attack) an entity is. For a given entity, e_i, exposure is computed as given in (2).

$$Exposure(e_i) = \frac{urc(e_i)}{trc(e_i)} \qquad (2)$$

In (2), $trc(e_i)$ is the total number of red capabilities that threaten the type associated with entity e_i, and $urc(e_i)$ is the number of those red capabilities for which no blue capability has currently been mapped, as identified in the environment description given in the Environment model. The Exposure method retrieves both

red and blue capabilities by entity type from the Reference Catalog model.

The set of red capabilities that threaten a given entity type, $trc(e_i)$, is the union of the red capabilities directly mapped to the entity type and the red capabilities of the entity type's parent, recursively up to the root of the taxonomy. Figure 6 illustrates the concept with an example. On the left hand side of the figure, the red capabilities for a given CISCO router consists of the red capabilities tied directly to the router plus those for parents leading to the root (CISCO Router, Router, and Network Device).
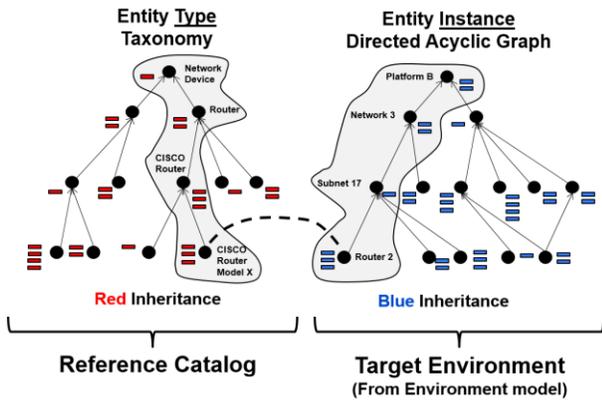

Figure 6 – Determining Exposure

The figure also highlights the fact that BluGen can be useful early in the system development process when the architecture is only known at a more abstract level. In such cases, many asset types are identified in generically terms (e.g., "router") since design details have not yet solidified. Nonetheless, threats can still be assessed using the same process, but starting from the more generic asset types closer to the root of the entity type taxonomy.

The right side of Figure 6 illustrates how the Exposure method gathers blue capabilities for a given asset instance. The Environment model allows one to specify an optional parent for a given entity instance. For example, the parent of a workstation might be the segment of an enterprise network. So, in a manner analogous to determining red capabilities for an entity type, the blue capabilities tied to a given entity instance are the union of the blue capabilities directly mapped to the entity instance joined together with the blue capabilities of the entity's parent(s), recursively. Unlike the Reference Catalog entity taxonomy, relationships in the environment form a directed acyclic graph.

Criticality. The Criticality method computes the worst case mission impact/criticality if an entity is attacked. An entity is more mission-critical if a greater

number of highly weighted missions rely on the entity and a greater number of high criticality data types are processed there. An overall asset's criticality is computed from a set of incoming criticality scores supplied from the Environment model.

To capture this notion mathematically, we compute raw criticality, $rc(e_i)$, for each entity e_i, as given in (3), where $mw(m_j)$ is the mission weight for mission j, and $crit(e_i, m_j, d_k)$ is the assigned mission criticality, a value between 0 and 1 inclusive, for the triple (e_i, m_j, d_k) that comes in as input to BluGen from the Environment model.

$$ rc(e_i) = \sum_{j=1}^{|M|} \sum_{k=1}^{|D|} mw(m_j) \cdot crit(e_i, m_j, d_k) \quad (3) $$

The final criticality/mission impact for a given entity, $fc(e_i)$, is the ratio of the raw criticality for the entity divided by the maximum raw criticality found across the entire entity set (4).

$$ fc(e_i) = \frac{rc(e_i)}{\max(rc(e_m))} \quad (4) $$

Thus, criticality values are expressed as a percentage of the maximum entity criticality found in the environment description given in the Environment model.

Mitigation. The Mitigation method computes the recommended set of mitigations, entity-by-entity, required to bring risk down to an acceptable level, as specified by the Environment model. For a given entity instance in the Environment model, we use the Reference Catalog model to look up the red capabilities that map to the entity based on its type. We then look up the blue capabilities that mitigate the red capabilities. If we call the resulting set M1 and we define set M2 as the blue capabilities currently mapped to the entity instance based on the environment description, then the missing mitigations are just M1 − M2. The set M3 defined by (c | c ∈ M2 and c ∉ M1) represents superfluous blue capabilities that are currently mapped to the entity instance but are not useful with respect to the threat.

Instantiation

Figure 7 presents an architectural view of the BluGen proof-of-concept instantiation. In this figure, each method described earlier is realized through a corresponding analytic (e.g., the Risk Method is realized via the "Risk Analytic" in Figure 7). The instantiation currently uses synthetic data for both the Environment and the Reference Catalog models. The front-end of the instantiation is implemented using Java and JavaFX [42].
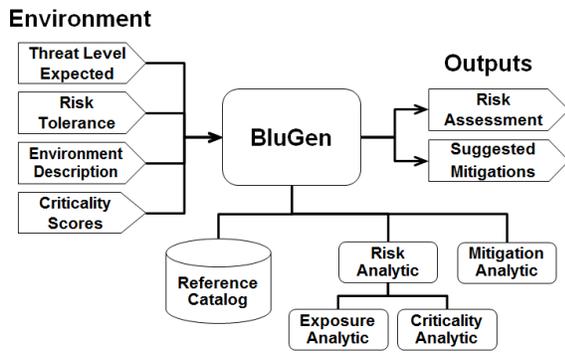
Figure 7 – BluGen Instantiation

Figure 8 shows a sample risk plot produced by BluGen. The plot is typical of a target mission/system environment early in its lifecycle before mitigations have been tied into the architecture. Thus, mean exposure levels tend to be high.

As Figure 8 shows, the data points represent entity instances from the environment description, not individual cyber attacks, as are traditionally represented on risk plots. Note that entities can include not just hardware and software assets, but systems, networks, the roles people play, and data types.
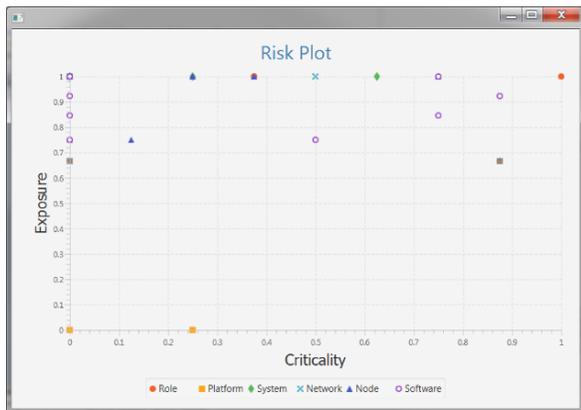

Figure 8 – Sample Instantiation Output

Similarly, Figure 9 shows a sample mitigation recommendations report automatically generated by BluGen. For each entity instance, the report shows the asset id and name, computed criticality and exposure scores, recommendations on which mitigations (blue capabilities) to add, and mitigations that could be removed because they do not map to any threat (red capability) per the Reference Catalog model. A given score is shown in red typeface if it exceeds the corresponding risk tolerance.

Mitigations in the Add column are missing blue capabilities (mitigations) that are required to fully mitigate the anticipated threat, as defined by red capabilities tied to the asset type of the asset named on each row. The Remove column identifies blue capabilities currently tied to the asset that do not contribute to the anticipated threat; thus, one might consider removing such blue capabilities to reduce overall system complexity and attack surface and to potentially reduce costs.

Figure 9 – Sample Mitigation Recommendations

5. Evaluation

Successful evaluation of BluGen is achieved by demonstrating the degree to which BluGen meets the requirements in Table 1. Before proceeding further, we state what is perhaps obvious: that BluGen is dependent on the correctness of the Reference Catalog and Environment models, the content of which is, at least in part, SME-determined. SMEs must populate the catalog contents with reasonable red/blue capabilities, entities, and related mappings. This section assumes that SMEs have populated the Reference Catalog with "correct" content, meaning that the relevant entity types, red and blue capabilities, and required mappings are in place. Table 3 describes how BluGen meets the first seven requirements from Table 1.

Table 3 – Meeting Requirements

ID	How Met by BluGen
R1	BluGen cross references red (threat) and blue (mitigation) capabilities with each other and to entities in the environment. With this information, BluGen computes an exposure metric, which is the ratio of unmitigated threats to the total number of threats that map by entity instance.
R2	Based on the criticality inputs, BluGen rolls up and computes mission criticality for each non-mission entity; that is, assets, roles, and data. Criticality is a measure of mission consequence if the entity is attacked.
R3	For each entity analyzed, BluGen plots the asset on a risk plot based on exposure and criticality scores.
R4	BluGen recommends mitigations for entities that score outside the risk tolerance region of the risk

ID	How Met by BluGen
	plot. Mitigations are based on reducing the entity exposure score to a level below the corresponding risk tolerance parameter.
R5	BluGen can analyze environment descriptions that reflect high level descriptions of entities typically known at the concept phase of a new program. BluGen can also analyze detailed Environment model descriptions typical of existing, deployed systems. The only requirement is that the entity types from any such models are mapped into the Reference Catalog.
R6	Holding constant both models (Environment and Reference Catalog), we assert repeatability of BluGen results based on the fact that BluGen's method implementations as algorithms will mechanically produce the same outputs given the same inputs. We do not analyze whether the same SSE team using BluGen to analyze the same model data on two different occasions will produce a repeatable result, as the focus of this paper is on BluGen itself. However, our (untested) hypothesis is that the results would be more repeatable compared to having the same team repeat an SSE analysis without the benefit of BluGen automation. This belief is based on the fact that the SSE team will have less analysis to do because of the substantial head start that BluGen offers. We make a similar argument for reproducibility. Variability in the results of two different teams analyzing the same target environment with the same Reference Catalog will be due to team differences.
R7	The entity type taxonomy in the Reference Catalog model (Figure 6) is general enough to accommodate traditional IT assets as well as specialized asset types found in industrial control settings. Over time, the entity type taxonomy is expected to expand in depth and in breadth, to accommodate new types of entities encountered in the different environments where BluGen is employed.

Our plan for evaluation of requirements 8 and 9 requires execution of an experiment. The experiment is based on a two-group, posttest-only, randomized experimental design [44]. While the experimental design has been articulated, space considerations prevent full description in the current paper.

Evaluation per Hevner DSRM Guidelines

Table 4 briefly summarizes BluGen in terms of the guidelines from Hevner.

Table 4 – Design Science Research Guidelines

Guideline	Discussion
1 - Design as an artifact	BluGen is described in terms of an interrelated set of designed artifacts.
2 - Problem relevance	Frameworks such as RMF now require organizations within the US government to assess and manage cyber risk to missions. We expect the number of required assessments to grow non-linearly for the foreseeable future, thus making purely manual SSE untenable.
3 - Design evaluation	We have evaluated 7 of 9 requirements with 2 requirements pending.
4 - Research contributions	Our contribution is the BluGen framework and related artifacts, which address the requirements in Table 1
5 - Research rigor	BluGen builds upon attack-centric risks assessment frameworks, as exemplified by NIST 800-30.
6 - Design as search	We considered various approaches (e.g., vulnerability, attack, and capability), and we factored out common steps carried out by SSEs that do not change from assessment to assessment.
7 - Research communication	This paper is a first step in communicating BluGen.

6. Discussion

As of this writing, the experiment mentioned in the previous section has not been executed. A future paper will describe and discuss experimental results.

It has not escaped notice that BluGen, while intended initially to serve the defensive cyber community could, in a trivial reformulation, be of use to those performing red team and penetration testing duties. In a reformulated BluGen (i.e., "RedGen"), the perspectives are flipped, so, for example, the risk plot of BluGen becomes an "attack attractiveness" plot, with the exposure axis relabeled as the "opportunity" axis, as the plot highlights entities that have high exposure and thus represent potentially easy opportunity for attacks for high impact attacks.

7. Conclusion and Future Work

In this paper, we described BluGen, an analytic framework that generates risk plots and recommends prioritized mitigations for a target mission/system environment based on a stated level of threat and risk tolerance.

Assuming the overall approach passes experimental validation, a possible future direction could be to prepare the BluGen Environment model input via automated means (today, model population is typically carried out manually). Both system and mission mapping techniques would be required.

The Reference Catalog mappings in the current BluGen instantiation were created by a small SME

team at Johns Hopkins APL. A future goal is to allow the larger cybersecurity community to peer review, update, and, most importantly, reuse the Reference Catalog as a shared community resource. Automated generation of mappings within the Reference Catalog is another area of possible exploration, using, for example, a supervised machine learning approach.

Currently, the mapping of mitigations to threats allows for a single "solution" to mitigate a given set of threats. An enriched catalog could capture multiple alternative solutions, that is, different mitigation approaches to address the same set of threats. Each solution could carry with it distinguishing attributes, such as estimates of solution acquisition cost, operational cost, mission performance cost, implementation complexity, and strength. This setup would then allow an automated form of trade-space analysis based on a given set of attribute value inputs.

8. Acknowledgements

9. References

[1]   D. W. Hubbard, *The Failure of Risk Management: Why It's Broken and How to Fix It*. Wiley, 2009.

[2]   R. Falkowitz, "The Automation of Risk Assessment," 2013. [Online]. Available: https://www.3cs.ch/the_automation_of_risk_assessment.

[3]   NIST, "Guide for Applying the Risk Management Framework to Federal Information Systems," 2010.

[4]   DoD, "Department of Defense Instruction Number 8510.01 - Risk Management Framework (RMF) for DoD Information Technology (IT)," 2014. [Online]. Available: http://www.dtic.mil/whs/directives/corres/pdf/851001_2014.pdf.

[5]   "Open Vulnerability Assessment System (Open VAS)," 2016. [Online]. Available: http://www.openvas.org/about.html.

[6]   C. Flanagan, K. R. M. Leino, M. Lillibridge, G. Nelson, J. B. Saxe, and R. Stata, "PLDI 2002: Extended Static Checking for Java," *SIGPLAN Not.*, vol. 48, no. 4S, pp. 22–33, Jul. 2013.

[7]   DISA, "Enterprise Mission Assurance Support Service (eMASS)," 2016. [Online]. Available: http://www.disa.mil/~/media/Files/DISA/Fact-Sheets/eMASS.pdf.

[8]   N. R. Mead and T. Stehney, "Security Quality Requirements Engineering (SQUARE) Methodology," *SIGSOFT Softw. Eng. Notes*, vol. 30, no. 4, pp. 1–7, May 2005.

[9]   National Institute of Standards and Technology, "National Institute of Standards and Technology 800-30: Guide for Conducting Risk Assessments," 2012.

[10]   "ISO/IEC 27001:2013 - Information technology, Security techniques, Information security Management systems, Requirements," 2013.

[11]   Object Management Group, "Unified Modeling Language (UML)," 1999. [Online]. Available: http://www.uml.org.

[12]   National Institute of Standards and Technology, "Framework for Improving Critical Infrastructure Cybersecurity," 2014.

[13]   R. Ross, M. McEvilley, and J. C. Oren, "DRAFT National Institute of Standards and Technology (NIST) Special Publication 800-160: Systems Security Engineering An Integrated Approach to Building Trustworthy Resilient Systems," 2016.

[14]   R. Dove, B. Wilson, and K. Kepchar, "INCOSE Systems Security Engineering Working Group," 2016. [Online]. Available: http://www.incose.org/ChaptersGroups/WorkingGroups/processes/systems-security-engineering.

[15]   National Institute of Standards and Technology (NIST), "FIPS Pub 199 - Standards for Security Categorization of Federal Information and Information Systems," 2004.

[16]   "CNSS Instruction No. 1253 - Security Categorization and Control Selection for National Security Systems, Version 2," 2012.

[17]   "Department of Defense Instruction 8510.01 Information Assurance Certification and Accreditation Process (DIACAP)," 2007. [Online]. Available: http://www.acqnotes.com/Attachments/DoD Instruction 8510.01.pdf.

[18]   S. Orfei, T. Leach, J. King, L. Mauro, and J. Fitzsimmons, "Payment Card Industry Security," 2006. [Online]. Available: https://www.pcisecuritystandards.org/pci_security/.

[19]   J. Gosler and L. Von Thaer, "Resilient Military Systems and the Advanced Cyber Threat," 2013.

[20]   NIST, "National Institute of Standards and Technology 800-39 - Managing Information Security Risk," 2011.

[21]   T. Weller, "Automation: The Key to More Effective Cyberrisk Management," *RIsk Management Monitor*, 2015. [Online]. Available: http://www.riskmanagementmonitor.com/automation-the-key-to-more-effective-cyberrisk-management/.

[22]   C. Knez, T. Llanso, D. Pearson, T. Schonfeld, and K. Sotzen, "Lessons Learned from Applying Cyber Risk Management and Survivability Concepts to a

Space Mission," in *IEEE Aerospace*, 2016.

[23] T. Llanso and P. Engebretson, "A Unified Model for System Security Engineering," in *Hawaii International Conference on System Sciences*, 2016, p. 8.

[24] K. Scarfone, M. Souppaya, A. Cody, and A. Orebaugh, "NIST Special Publication 800-115: Technical Guide to Information Security Testing and Assessment," 2008.

[25] C.-W. Ten, C.-C. Liu, and G. Manimaran, "Vulnerability Assessment of Cybersecurity for SCADA Systems," *Power Syst. IEEE Trans.*, vol. 23, no. 4, pp. 1836–1846, Nov. 2008.

[26] Tenable, "Nessus Vulnerability Scanner," 2013. [Online]. Available: http://www.tenable.com/products/nessus.

[27] "Network Mapper (NMAP) Reference Guide," 2014. [Online]. Available: http://nmap.org.

[28] T. (JHU/APL) Llanso, "Working Paper: Estimating Latent Vulnerabilities," 2016.

[29] National Vulnerability Database, "Vulnerability Summary for CVE-2014-0160 (Heartbleed Vulnerability)," 2014. [Online]. Available: http://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-0160.

[30] NVD, "Vulnerability Summary for CVE-2015-8642," 2015. [Online]. Available: https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2015-8642.

[31] T. Llanso, G. Tally, M. Silberglitt, and T. Anderson, "Applicability Of Mission-Based Analysis For Assessing Cyber Risk In Critical Infrastructure Systems," in *International Federation for Information Processing (IFIP) - Critical Infrastructure Protection VII*, 2013th ed., vol. VII, Springer Berlin Heidelberg New York, 2013, pp. 135–148.

[32] Mitre, "Cyber Mission Assurance Engineering: A Risk-Based, Threat-Informed Approach to Address Advanced Adversaries," 2013.

[33] S. Musman, M. Tanner, A. Temin, E. Elsaesser, and L. Loren, "Computing the Impact of Cyber Attacks on Complex Missions," *2011 IEEE Int. Syst. Conf.*, pp. 46–51, Apr. 2011.

[34] T. Llanso and E. Klatt, "CyMRisk: An approach for computing mission risk due to cyber attacks," in *IEEE International Systems Conference. Ottawa*, 2014, pp. 1–7.

[35] B. A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004.

[36] V. Vaishnavi and B. Kuechler, "Design Science Research in Information Systems," *Design science research in information systems and technology*, 2011. [Online]. Available: http://desrist.org/desrist/content/design-science-research-in-information-systems.pdf.

[37] A. R. Hevner and S. Chatterjee, *Design Research in Information Systems*. 2010.

[38] JHU/APL, "An Introduction to Mission Information Risk Analysis (MIRA)," 2015.

[39] T. Llanso, A. Dwivedi, and M. Smeltzer, "An Algorithm to Estimate Cyber Attack Level of Effort," in *IEEE International Systems Conference, Vancouver*, 2015, p. 6.

[40] DoD, "DoD CIO/AT&L Capability-based Threat Model," 2015.

[41] "National Institute of Standards and Technology Special Publication 800-53 Revision 4," 2013.

[42] Oracle Corporation, "JavaFX," 2016.

[43] D. Jones and S. Gregor, "The Anatomy of a Design Theory," *J. Assoc. Inf. Syst.*, vol. 8, no. 5, 2007.

[44] W. Trochim and J. Donnelly, *The Research Methods Knowledge Base*. Atomic Dog, 2008.