

Securing the Smart Grid

Southern California Edison
Advanced Technology

1 Introduction

Southern California Edison (SCE) seeks to discover, evaluate, and adopt energy and information technologies to implement a smarter, more robust electricity infrastructure to deliver greater amounts of renewable generation, use electricity as a fuel for vehicles, enable consumers to become active participants in the energy supply chain, and ensure the continued reliability and vitality of our nation's energy economy.

As such, SCE believes that many aspects of a smarter grid will need to be operational by the year 2020 to enable California's ambitious policy goals, such as AB 32 Green House Gas reductions, Zero Net Energy homes, California Solar Initiative, Advanced Metering Infrastructure, California's Renewables Portfolio Standard, Low Carbon Fuel Standard, and wide-spread consumer adoption of Plug-in Electric Vehicles. By the year 2020, SCE aims to modernize the grid to deliver a cleaner energy supply from renewables and integrated distributed resources, energy-smart consumer devices, and electric vehicles while improving reliability, safety, and cost-effectiveness.

Within this context, SCE's overall smart grid strategy encompasses five areas that address a broad set of requirements to better position SCE to meet both current and future power delivery challenges:

- Renewable and Distributed Energy Resources Integration – Integrate and manage new sources of renewable and distributed energy supply.
- Grid Control & Asset Optimization – Improve capital efficiency and asset utilization using energy storage, new materials, and better intelligence and technology for optimal system management.
- Workforce Effectiveness – Maximize workforce safety, productivity, and effectiveness by using enabling tools and technologies.
- Smart Metering – Enable the grid to automatically adjust to changing loads and supply requirements.
- Energy-Smart Customer Solutions – Empower customers to become “active” participants in the energy supply chain by, in part, managing their own energy consumption; and provide customers with the means to use electricity as a fuel for vehicles.

These five areas define our business objectives, and are built on a foundation of new telecommunications networks, software systems and other information technology. The smart grid will be realized through innovations in both energy technology and information technology. Consistent with the U.S. Department of Energy's and National Energy Technology Laboratory's Vision for the Modern Grid, SCE's smart grid will enable the increase of intermittent and renewable resources (such as wind and solar power) and spark greater use of plug-in electric vehicles by increasing system flexibility; reduce greenhouse gas emissions;

avoid the economic losses associated with catastrophic failures and wide-area blackouts; foster energy conservation, energy efficiency and demand response capabilities by providing customers with better energy use information and choices; reduce operating costs and improve reliability and safety by providing real-time information for system monitoring and system automation; improve maintenance and operations practices on the electrical grid; and facilitate the development of a “Clean Tech” economy.

The electric grid is and will be a national security asset and therefore, cyber-security is a significant consideration in the development of a smart grid. SCE recognizes the need to increasingly safeguard from cyber-attack the communications and computing systems installed throughout its grid network. To tackle the cyber-security challenge, SCE has pioneered a profound and compelling approach to addressing complex systems security engineering. This paper describes the approach in a broad slice from beginning to end, and illustrates the benefits that may be received from innovative application of well-understood engineering disciplines. In partnership with ten other utilities nationwide, the U.S. Department of Energy, and Carnegie Mellon University, SCE provided this approach contributions to the recently completed first element for a secure advanced metering infrastructure.

1.1 Problem Statement

The smart grid as envisioned by SCE, expanding on the Energy Security and Independence Act of 2007 to support other future utility and customers needs, requires a new layer of information technology that dynamically links most all elements of the grid and interconnected devices into a unified network. The creation of such a network enables significant benefits, but also introduces potentially significant cyber-security threats.

1.1.1 Poor Definition of Problem Space

How does one define security for the smart grid? What exactly are the objectives and desired outcomes for such a lofty goal? The term “smart grid” has been used by many to articulate a vision of the next generation of the electric utility. While the visions expressed are generally, in at least loose alignment, the differences in subtlety and interpretation are almost as numerous as the invocations of the term. In many ways, this illustrates a healthy, innovative and creative environment – a good thing when it comes to solving complex problems such as transforming the electric power industry. However the “devil in the details” cliché comes back to haunt the security-minded when one tries to do one of the first essential tasks: scope and define the problem.

When the security engineer starts these tasks, they first and foremost strive to understand what it is they are trying to protect. In order to succeed in the business environment, this understanding must align with business values. However, business is all about streamlining processes and removing hurdles. Security must shed its dogged image as the inhibitor and take the role of the enabler. Yet enabling business functions depends upon a solid understanding of what those business functions are. For each utility we must clearly articulate the business functions to be secured, else the problem of security for the smart grid will be nebulous, elusive and open-ended for all but the most sophisticated approaches.

1.1.2 Volatility and Evolution in Target and Landscape

The ability of the smart grid to elude universally crisp definition reflects one of the more difficult problems in modern security: the rapid rate of change for emerging markets and technology. Volatility and evolutionary rate represent exponential functions of technological novelty, and bear remarkable similarities to basic physics. Things must have mass in order to have inertia, and the newer a technology is, the more likely it is to disappear (or become subsumed) as quickly as it appeared. Technology needs time in order to accumulate sufficient mass and start the journey to stability.

Instability has a direct and immediate impact on the ability to provide security. A moving target is much more difficult to secure than a stable one simply because the security controls must accommodate this motion and variability. By comparison, it is much easier to develop the initial functionality than it is to understand all the implications of the functionality, which is in turn easier to understand than the details of the implementation. Yet security depends on understanding all of the implications and details. Researchers find more holes in code as time goes on because time allows for this understanding. Time is the friend of security, and by definition is the one resource not yet available with the smart grid.

1.1.3 Multiple Interests and Moving Pieces

One of the causes for differing interpretations of the smart grid is the number of stakeholders and the variety of their interests. The smart grid has many interested parties, both inside and outside the utility. Inside the utility, interest varies with both proximity to and role within the smart grid. Designers, builders, maintainers, consumers, and sponsors all have vested interests and each has a differing agenda. Outside of the utility the agendas diverge even further as one considers entities such as regulators, customers, retailers, and vendors. The number of stakeholders and concerns around the smart grid is testament to its impact – only such a broad topic could have this kind of effect.

For the security engineer this effect complicates the equation. Multiple interests often bring conflicts, and differing agendas have the potential to place the security engineer in the middle of an unpleasant disagreement. Requirements become further complicated when each of these parties have their own pieces for which they are responsible. The end result is a landscape in continuing and varying degrees of flux, with each element on its own vector. The environment could hardly be more challenging.

1.1.4 Novelty of Application and Domain

As if a challenging environment were not enough, the smart grid provides at least one more point of consideration. The electric power industry is not known for leading-edge innovation and the rapid adoption of new technologies. Yet this kind of fundamental transformation is precisely what the smart grid is all about. Cultural issues abound as traditionally conservative engineers wrestle with placing unproven technology into service and depending on its operation to compete in the modern world. The security engineer must account for this paradigm shift and in many cases lead the utility through discoveries that could be pleasant or highly undesired.

1.2 Risk Management and Systems Engineering

In the midst of all the challenges of the smart grid, SCE has forged a profound yet surprisingly practical approach. The thesis is this: security for novel technological problems may be addressed through

disciplined application of both risk management and systems engineering principles and methodologies. This paper will illustrate how the use of these very tangible tools has the potential to transform the way the utility world addresses issues of security from this day forward.

1.3 Approach

The risk management and systems engineering approach is comprised of three phases of work that together form an arc for problem solving in the complex environment. This approach is defined by the phases of reduction, decomposition and direction. Execution of these phases provides the utility with a solid understanding of the problem space, comprehension of drivers and tendencies, and ultimately a means to control and navigate the territory.

1.3.1 Reduction: Bounding, Grounding, and Profiling of Variables

In the first phase of the approach the utility bounds the problem by focusing on scope definition, identification of objectives, and function definition. The utility then grounds the problem by nominating commonalities and parallels in understood domains, identifying differences and discrepancies, and evaluating relationships and reusable material. The utility concludes the first phase by profiling the remaining variables for depth and impact, magnitude of uncertainty and unpredictability, and the nature and drivers for associated variability.

1.3.2 Decomposition: Modeling, and Risk Driven Requirements Engineering

The second phase of the approach focuses on decomposing the problem space through exploration and extraction. The utility explores the problem using the modeling techniques of contextual analysis for the environment and interactions and conceptual analysis for data, processes, and communications. The utility is then prepared to extract requirements from the space by examining functional and non-functional assurances, dependencies between elements, and the cost of complexity in potential solution paths. Risk management provides the ability to correlate value to need, record assumptions and dependencies, and mitigate technological uncertainty. In this phase requirements become a tangible output of the risk and architectural analysis.

1.3.3 Direction: Maturity Models, and Motivator Identification

The last phase of the approach determines the direction for the utility by leveraging the requirements management tools and defining maturity and capability models. The utility produces maturity and capability models embodied in evaluation criteria for products and technologies, formulas and links to allow for adjustment of the model, requirements for the procurement process, and documented implications that will drive guidance for operations and maintenance. Finally, the utility looks at the ability to reuse components from previous models as well as the potential for components from the model under development to be used again in future problem sets. When necessary, the utility uses the reduction and decomposition phase to motivate design.

When done properly this phase results in the utility's ability to defend its choices and actions, and to provide strong traceability in the event that parameters and assumptions change enough to warrant re-examination of the problem space. This paper also examines the alignment of high-level motivators in the process with overall business priorities.

2 Reduction: Bounding, Grounding, and Profiling of Variables

The reduction phase focuses on containing the problem space to a form that can be attacked in a disciplined manner with deterministic results and no superfluous issues. The steps normalize and format the problem characteristics into meaningful variables, providing the utility a solid understanding of the issues at hand and laying the groundwork for structured analysis and decomposition.

2.1 Bounding the Problem Space

The first step to this approach places conceivable bounds on the problem space. Limiting and defining the issues at hand is critical to the success of the overall analysis.

2.1.1 Scope and Points of Demarcation

The scope of the problem is defined as the exhaustive set of elements that are subject to a degree of influence by the security engineer for a given domain. A tighter scope implies fewer variables and accordingly less complexity; therefore, the security engineer must recognize the significance of a tight scope and strive for that goal to the extent it remains practical.

Frequently however, and especially in the case of the smart grid, the scope may not be reduced to something simple due to the interwoven nature of smart grid functionality. In such cases the security engineer must look for scoping opportunities with low numbers of external interfaces. These opportunities may typically be found at a business function boundary, which presents a synchronistic opportunity to align the scope of the security problem with that of the business. The benefits of this alignment are significant, as it allows the security engineer to produce architecture, policies and specifications that can be dovetailed with the established business areas.

Once a scope has been selected the security engineer walks the logical and physical perimeters, noting points of demarcation in the architecture where communication paths cross the perimeter. These points of demarcation should be located on one side or the other of system elements for clarity. Scoping lines through the middle of singular elements are unhelpful and a harbinger of contention. Each line that crosses the perimeter must then enter a neighboring system, which the security engineer catalogues in terms of name, control and ownership.

The security engineer first defines boundaries in the system as logical segmentations. In traditional security engineering these segmentations take the form of segmented communication networks. However with the advance of virtualization and finer grain security controls, the engineer must move beyond simple network parameters and look at all aspects of the information lifecycle. Using this principle security boundaries and domains are better defined as a collection of information and resources that can be grouped by common characteristics.

2.1.2 Objectives and Statements of Value

The security engineer must clearly understand and capture the high-level business objectives for all elements within the defined scope as well as neighboring logical systems. The security engineer examines these objectives and attempts to rate the relative value of a given element. For this discussion, assets can

be either resources or information. This practical risk driven approach is then adaptable to all business operations (i.e., Risk Adaptive Security).

The risk adaptive approach allows the security engineer to make appropriate decisions about mitigation techniques based on the cost-benefit trade-offs of various controls. However as we will show later, the determination of benefit is rooted in a solid understanding of asset values and the nature of their sensitivities. The security engineer must therefore capture statements describing the value of business functions in the context of the overall business. This value may take the form of either a monetary amount or a relative portion of criticality to higher level functions or value streams. The security engineer documents the characteristics and later uses them in risk management calculations.

The security engineer must be cautious not to fall into the trap of assigning value to system elements based on criteria such as replacement cost, and subsequently composing overall system value based on the sum of system elements. In complex problem sets such as the smart grid, elements are frequently used for more than one purpose and serve more than one business objective. Element-centric value statements will distort this picture and present opportunity for elements to be dramatically undervalued. The result will be an inaccurate risk model, exposing the business to hazards of individual component failure.

2.1.3 Functions and Definitions

In the context of risk management, individual components are a means to an end. Protecting a server is important, but that importance varies greatly depending upon the operational and business context of the machine. The machine's protection takes on different priorities if the system it supports is essential to the operation of the business as opposed to providing supplementary information for strategic analysis. The business function is what links the individual component or element to the highest-level value stream.

At this phase, the security engineer is heavily dependent upon process-related documentation associated with the business functions. Thoroughly defining the process forms a foundation for understanding interdependencies and component-level criticality. The documentation must tell a complete story with no holes and verify against the problem scope. Processes may only enter and leave the scope through the aforementioned points of demarcation.

The security engineer cross-checks all processes for common elements and either references an external source for a glossary or develops one specific to the problem scope. An external source is preferred if it is authoritative and does not conflict with other relevant documentation. Conflicts in definition must be resolved before meaningful work can progress.

2.2 Grounding to Understood Domains

Grounding the problem to understood domains provides meaning to the utility in an otherwise untethered environment. The activity results in tangible approach points for utility engineers and allows the engineer to constrain, separate and solve for unknowns.

2.2.1 Nomination of Commonalities and Parallels

One of the more fortunate aspects of the smart grid is the relative rarity of completely novel technology. The vast majority of new applications to the utility are in fact a re-mapping of existing technologies into a

previously-understood space. The security engineer must leverage both ends of this equation and pull lessons from the history of each technology as well as knowledge embedded in the application space.

Frequently the security engineer may also find corollaries to other endeavors where existing, but possibly different technology was applied to another understood space. These instances are best found within the utility world, as the translation of one context to another tends to lead assumptions that may or may not hold.

The security engineer records each found instance of a commonality or parallel to other implementations, describes the nature of the relationship to the extent of understanding, and notes any implications drawn against the current problem set.

2.2.2 Identification of Differences and Discrepancies

For each of the instances of commonality or parallel found, the security engineer must be diligent in noting the differences from the current problem set. These differences represent cautionary flags for translation of assumptions from one problem space to another.

The obvious implications of differing contexts are relatively easy to discover. However the subtle primary implications and secondary/tertiary/etc. (ripple effect) implications are much more difficult and may require the expertise of someone familiar with both domains. Ideally the expert has made a career transition between the domains and is able to speak directly to the contextual shift, but this expertise is not common. In such cases the utility must evaluate the effort required to gain this expertise against the potential value of the knowledge gained from the commonality or parallel.

Note that the inability to evaluate the change in context undermines the found relationship, and ignoring the contextual change poses serious risk to the integrity of the overall process. The safe decision in this circumstance is to not use the relationship in the model, and therefore sacrifice potential lessons learned in the interest of maintaining the integrity of all assumptions.

2.2.3 Evaluation of Relationships and Reusable Material

Once the security engineer has catalogued relationships to other (i.e., previous) problem spaces and noted all contextual differences, the engineer must evaluate the strength of the relationship and value of lessons to be drawn. The strength of the relationship and value of lessons are two separate factors and must not be confused. Rather the two factors serve as multipliers in determining the ability to draw from lessons learned – to reuse material, assumptions, and/or conclusions from the previous problem space.

In one of the trickier exercises in the overall process, the security engineer must endeavor to be as objective as possible in evaluating the relationships. One must accept the premise of an arbitrary, absolute scale where the perfect result is a direct corollary to a virtually identical problem space within the same industry or company, and the worst result is a parallel to a highly dissimilar problem space in a different industry with many discrepancies. The security engineer must evaluate each relationship against this theoretical scale and assign a level of faith to the relationship. Use of a relative-only scale does not account for possible statistical anomalies, especially in small result sets, and exposes the process to the risk of placing either too much or too little faith in the relevance of a relationship.

2.3 Profiling of Remaining Variables

At this point in the process, the problem space will likely have a number of variables that have yet to be addressed. As the last part of the foundation work, the security engineer addresses these remaining variables in exercises to understand their role in the problem space.

2.3.1 Depth and Impact (Relevance and Pervasiveness)

The security engineer evaluates each unknown in the problem space for its relative proximity to the core of the problem and frequency of occurrence in the space. This procedure will be familiar to those experienced in risk management, as there is a direct corollary to severity and frequency aspects of threat analysis. Likewise, the depth and impact analysis will result in a prioritized set of variables to be addressed by the security engineer.

2.3.2 Magnitude of Uncertainty and Unpredictability

In addition to a priority, each variable will also exhibit degrees of uncertainty and unpredictability. Uncertainty and unpredictability are the antithesis of accuracy and precision, respectively, and should be addressed in similar manner. These properties can often be quantified further in detailed discussions with domain subject matter experts.

2.3.3 Characterization of Variability and Associated Drivers

While depth, impact, uncertainty, and unpredictability are an adequate set of characteristics for a variable, the security engineer must not stop the analysis. Frequently variables may be further profiled in terms of behavior, and this profile provides significant insight into solving the overall equation. The security engineer must document behaviors of any consistency not captured in the previous characterizations. The scientific chemist's approach translates well here, as each behavior must be thoroughly described with no presumption regarding cause during the capture process.

The effective discovery of drivers for the variables depends upon the sequential separation of the behavior capture and analysis processes. Once all of the behaviors have been captured, the security engineer forms theorems delineating how various drivers may influence the variables and explains the behaviors. The security engineer should evaluate each theorem for completeness, integrity and correlation to other theorems. All behaviors should have at least one supporting theorem, as unexplained behaviors are flags of inconsistency and represent possible clues and keys to solving the overall equation.

3 Decomposition: Analyses and Requirements Engineering

The decomposition phase combines methods of pattern recognition and business process analysis to form a set of objectives that can be mapped to requirements. These objectives when linked to risks can be used in multiple aspects of the utility's overall security framework.

3.1 Domain Analyses

The domain analyses portion of the process requires the security engineer to step back from the minutia and focus on the identification of usage patterns at the system level. The security engineer will subsequently refine these patterns in an iterative process to form a set of security domains to be

referenced throughout the remainder of the work. While security objectives could be stated at a more granular level, the grouping of common elements can significantly shorten the process.

3.1.1 Contextual Analysis: Environment and Interactions

By the definition of scope and points of logical demarcation [above], the system under discussion does not exist in a vacuum. The relationship of the system to its surroundings represents an important aspect of understanding how the system must operate. All environments place certain constraints upon a system and these constraints influence if not dictate the interactions that cross the system border.

The environmental analysis is largely driven by issues of ownership, control and proximity. The security engineer must draw both physical and logical maps of the system and systematically characterize the spaces outside, but adjacent to the system. The starting point is somewhat arbitrary as distinctions between groups of external entities emerge through definition, analysis and description of each previous group in a self-feeding process. The security engineer should stop the process at the point where distinctions no longer influence the behavior of the system.

For each distinct space identified outside the system, the security engineer must define the nature of the interaction that crosses the system boundary. The security engineer defines the interaction through a basic description of the business function and associated security concerns such as confidentiality, integrity and availability. While the set of security concerns may vary depending on the reference model adopted, the security engineer must be certain to address each concern for each business function and provide an explanation even in circumstances where the concern does not apply.

3.1.2 Architectural Analysis: Data, Process and Communications

After examining the relationship of the system to its external neighbors, the security engineer decomposes the system under discussion in light of data, processes and communications. The security engineer must look at data in at least three basic phases: data at rest, data in transit and data in transformation. In each of these phases the security engineer applies a very similar analysis (as mentioned above) of business characteristic description with associated security concerns.

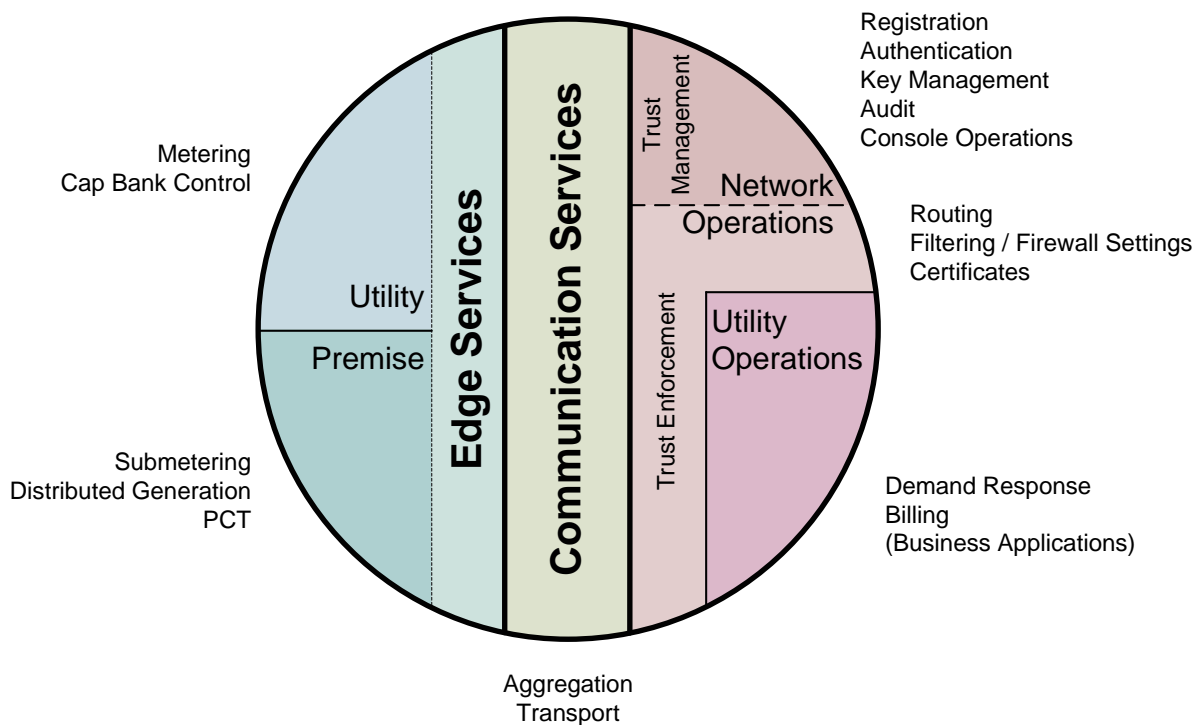
For process analysis, the security engineer must leverage available collateral from relevant business units in the organization such as Process Flow Diagrams and Use Case Narratives as a foundation. If sufficient collateral does not exist, this is an indicator that the organization may be out of synchronization in terms of maturity across functions and that a security process such as the one described herein may be at risk of compromise through organizational usurpation.

Many business processes are likely to have similar requirements and overlapping paths through the system from the security perspective. The security engineer should consolidate these paths for simplicity, but maintain a strong naming association and traceable lineage for each original business process analyzed. Again, the security engineer will describe how each of the defined security concerns relates to the entirety of each process as it transitions the problem space.

For communications analysis, the security engineer may not have the luxury of directly relevant business collateral. In the event such does not exist, the security engineer must consult the technologists in the

organization and catalog each potential technology and its application. This may be an iterative process as the security engineer translates technology-centric drawings and descriptions into the security domain space.

SmartGrid Security Domains



Text outside the circle represents example activity for service domains.

Figure 1 Smart Grid Security Domains

3.2 Risk-Driven Requirements Engineering

The requirements engineering field is assumed herein to be a well-developed and mature discipline; therefore, this document focuses on the application of requirements engineering specifically to the security space rather than discussion of the discipline itself.

For use in a security context, two forms of requirements generation prove fruitful. The first form uses existing requirements developed for other applications or borrowed from other industries. This form is also known as requirements “farming,” with the requirements subsequently filtered using the risk management process. Requirements may also be generated by following the risk-driven requirements process through to its natural conclusion, then translating resulting security objectives into requirements statements, herein called “natural generation.” The farming technique is typically the quicker option, although less precise. In either case the requirements are traceable to the risks in the system.

3.2.1 Farming

One of the more efficient means by which the security engineer can address requirements is to leverage the substantial body of institutional and regulatory material that already exists for relevant domains. The security engineer gathers this material and scours it for security-related requirements statements, documenting the statement itself, the source, and optionally one or more editorial categories (i.e., security concerns) to assist with organization. The security engineer then organizes and normalizes all statements, producing a master catalog of requirements from which they can pull and apply to appropriate need areas.

3.2.2 Natural Generation

Security engineers also have the option of following the risk process through to its natural conclusion. Risks are based on probability and impact, which are in turn based on asset value and threats to that value. If these are sufficiently defined then mitigations can be placed against these risk items. The purpose of this mitigation can be positively stated as a security objective, which may be subsequently translated into requirements.

In practice it is often necessary to use both requirements methods; used in conjunction farming and natural generation produce a comprehensive set of requirements.

3.2.3 Mapping

The security engineer systematically goes through each of the modeling diagrams and loosely applies requirements to each security domain, regardless of complexity or expense. Not all requirements will apply to each domain and the security engineer must exercise judgment in determining the relevance for an individual application. The risks associated with an error in judgment at this phase are mitigated in the resolution step below. This step allows for practical assertions such as, “system elements that do not have direct user interaction do not need special user interface related requirements.”

3.2.4 Risk Resolution

The basic function of risk management in this context is to provide the utility with an understanding of the financial relationship between asset values, risk exposure and risk mitigation techniques. Each mitigation technique provides a certain amount of coverage for some set of risk exposure. Risk exposure in turn correlates threats and vulnerabilities to asset values in the context of undesirable outcomes or loss/reduction of a value stream.

At a more detailed level, the security engineer drills inside of the value stream context and uses business process definitions to look at associated risks within defined security domains. (A security domain is a logical or physical area within which all security characteristics, requirements and constraints are consistent.) Domains have associated resources and information, both of which have value and are therefore sensitive to threat exposure. These threats have a certain probability of success. The security engineer uses this probability and value to determine a risk rating for both the domain and for individual elements within the domain.

Once the risk rating is established, mitigation objectives can be established for each one of the assets within the domain. These objectives are either mapped against existing requirements or used to generate new security requirements.

The security engineer again goes through each of the modeling diagrams, this time checking to see if all security concerns previously identified in the individual analyses (e.g., data, process, communications) have been addressed by at least one applied security requirement. In the event a security concern has not been addressed, the security engineer consults the developed catalog for a possible miss in requirement application. If the security engineer finds an appropriate requirement they apply it to the entire security domain, else they must compose a requirement to fit the needs of the situation.

4 Direction: Risk Management, Maturity Models and Motivator Identification

The direction phase brings together the analysis done in the first two phases and provides the utility with end products that will serve multiple business needs. These products align the security efforts with the overall corporate agenda.

4.1 Development Planning

Risk management tools link the analyses and requirements developed in the decomposition phase to the actionable plans the utility will use in the remainder of their systems development lifecycle. This link must be traceable in both directions, as traversing the link forward allows the utility to make decisions and traversing in reverse provides defense of choices and actions.

Once the security engineer has completed the decomposition phase, they are able to provide a list of candidate mitigation techniques that can be applied to protect the assets of the system under discussion. The security engineer must now formulate a specific development plan (i.e., protection profile) for the specific application domain or problem space.

4.1.1 Record of Assumptions and Dependencies

The security engineer must take care to identify engineering and business assumptions as well as system interdependencies when developing the protection profile. The description of the assumptions and dependencies is not new work; however the process of linking the source information is significant. The security engineer references the Functions and Definitions from the reduction phase and the Contextual and Conceptual Analyses from the decomposition phase and uses a bi-directional pointer or link to the relevant material, as changes or updates from an arbitrary point in the future must be translated across the model.

4.1.2 Mitigation of Technological Uncertainty

As the security engineer links the assumptions and interdependencies to the Protection Profile they must also reference the Profiling of Variables work done in the reduction phase. The variable profiles magnify uncertainty as it is projected through the analysis and the security engineer must react accordingly, noting where conclusions and determinations are either well-founded or poorly understood. In some

circumstances where the variable drivers are well-understood, a degree of uncertainty may be mitigated by escalating the level of protection. However the security engineer must exercise strong caution in these decisions as an unanticipated change in variable profile can quickly overwhelm or make irrelevant the technical controls selected.

Regardless of individual variable unpredictability, this approach provides the utility with a strong understanding of the risks involved in each action. The understanding alone allows the utility to make informed decisions and select a course that mitigates risk from technological development and uncertainty.

4.2 Maturity and Capabilities

The most visible products from this approach are the elements enabling the utility to effectively interact with external entities. Relevant tasks include communication of requirements to stakeholders (i.e., vendors), evaluation of products and technologies, adjustment of the security and/or architectural model, and guiding operations and maintenance procedures. The utility also reaps significant benefit when asked why or how it reached a certain conclusion as the documentation provides defensibility and traceability for all choices made.

4.2.1 Procurement Requirements and Operational Implications

The mapping of the protection profile against security domains [above] provides the security engineer with a discreet set of security requirements. These requirements may be directly shared with external entities such as vendors in a Request for Proposal (RFP).

Satisfaction of these requirements has direct impact on operational decisions and guidance. If some requirements are not met, the utility must either look elsewhere for patching solutions or develop new security solutions. Similarly, the utility may discover at some later date after implementation that products claiming to meet the requirements may not in fact do so, or that the requirements were not sufficient. In such circumstances the model will allow the utility to re-examine the situation and determine both root cause as well as best course of action.

4.2.2 Evaluation Criteria

In a direct counterpart to use in an RFP, the utility will use the requirements and supporting material in its evaluation of proposed solutions and products. While the requirements themselves provide the first level of criteria, the utility may augment the evaluation by determining further impact to the model from requirements that are not met.

The utility may also consider how the model is affected in the event one or more requirements are surpassed by varying degrees. It is possible for the utility to discover that one requirement falls short, but the model is still acceptable due to overcompensation in other areas.

4.2.3 Formulas and Links

At its most sophisticated level, the model developed by this approach may be represented by a complex series of formulas and links with parameters and determinants accommodating the need for research.

Each of the relationships described herein represents a strong association that may be mapped out mathematically allowing the security engineer to express utility desire and limitations through ramps and exponential curves. While this component of the approach is not essential, the utility benefits significantly from investment in its development. The utility may implement automation tools to update or change the model, or run multiple simulations for extraction of solution preferences or discovery of unanticipated options.

4.2.4 Defensibility and Traceability

Regardless of whether the utility implements automation in the model this approach provides the utility with strong traceability mechanisms linking business requirements all the way to security technology decisions. This traceability relieves the utility of dependency upon walking knowledge and allows the utility to defend its actions and choices to external entities. At any point in time someone completely unfamiliar with the history of the project can examine the model and understand how the utility reached each decision.

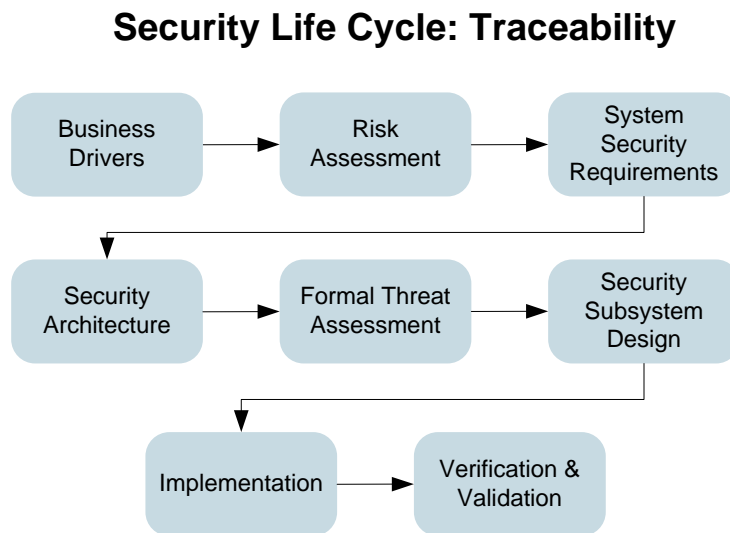


Figure 2 Traceability

4.3 High-Level Motivators

The ability to reuse components translates directly into a reduction in long-term level-of-effort for the utility. However these components must still allow for customization and specificity to be truly efficient and effective tools.

4.3.1 Antecedent and Subsequent Reuse

SCE has designed each of the products in this approach with a high degree of modularity. As the utility adopts this approach, it will accumulate a library of these analyses, requirements and risk assessments. For the utility to fully leverage this library it must maintain a conscious awareness in the early iterations

of the process, remaining mindful of how each module can be used again in the future. In future iterations the utility continually consults this library, looking for patterns with potential application to the new problem set.

The security engineer will ultimately assemble the analysis diagrams into a map of utility functionality. As each section of the map is developed, it becomes easier to delineate and describe subsequent sections, especially at the points of interface. The security engineer will reuse these interface definitions as well as the underlying structure of information to frame the new section. Each time this is done, the security engineer retains the information structure elements that are beneficial to long-term reuse, trims those elements that consistently do not add value, and adds elements where needed. These modifications to the information structure tune the library over time to provide greater service to the utility.

4.3.2 Customization and Specificity

As the security engineer defines and refines each work product information structure, they must keep in mind how applications of the product will vary over time. The key to allowing the products to adapt is to make moderately liberal use of parameterization. The parameters should provide the utility with the ability to tailor the products to new environments and accommodate changes in technological constraints.

These parameters should also account for the assumptions in the model to the extent possible, as these assumptions have some probability of changing over time. The security engineer should make a practice of reviewing these assumptions on a periodic basis, much like they do periodic policy reviews. The subtle erosion of engineering assumptions poses one of the more serious risks to information security, and may only be addressed through disciplined security practices. When the security engineer finds sufficient variation in the assumptions that would affect the resulting decisions, they must call for the utility to revisit the process and recalculate the risk equation.

5 Conclusion

This paper has presented a broad examination of the process SCE has developed to solve complex security engineering issues as it transitions to the smart grid. This approach has shown tremendous benefits to the utility in terms of work/product reuse, traceability and defensibility of actions, and mitigation of security and technology risks. Further, the approach provides tangible and provable assurances through quantification of unknowns and elimination of guesswork.

5.1 Summary of Approach

The three phases of reduction, decomposition, and direction provide the utility with a solid understanding of the problem space resulting in a traceable and reusable model with multiple high-value output work/products.

5.1.1 Reduction

The reduction phase of the approach bounds the problem to a defined space, grounds the problem to understood domains and reference points and profiles remaining variables. These actions shape the problem space such that it can be attacked with formal discipline and thus producing deterministic output, eliminating superfluous issues, and laying the groundwork for structured analysis and decomposition.

5.1.2 Decomposition

The decomposition phase combines methods of pattern recognition and business process analysis to explore the problem space and extract a model with associated descriptors. The utility uses modeling techniques for contextual analysis of the environment and interactions and conceptual analysis of data, processes and communications. Requirements are generated by examining risks associated with the domains.

5.1.3 Direction

The direction phase assembles the research from the first two phases and links it to end products that will serve multiple business needs. The risk management process provides the utility with the ability to trace its choices, defend its actions, and port knowledge to a tangible and highly reusable model capable of mathematical abstraction.

5.2 Determination

Disciplined application of both risk management and systems engineering principles and methodologies provides measured, traceable and defensible assurance in the face of novel technological problems. SCE has reaped tremendous benefits from the development of this approach, and associated work products have proven to be tangible validation of the hypothesis.

5.3 Application to Problem Space

While the processes described herein have been implemented by an electric power utility, SCE believes they have strong potential for application to many problem spaces that exhibit complexity and novel applications of technology. The characteristics of this approach make it particularly compelling for addressing the new generation of security challenges presented by the smart grid.

5.3.1 Volatility of Smart Grid Domain

The smart grid is currently an emerging front in the electric power industry. While the Energy Independence and Security Act of 2007 provides a list that represents the broad categories of applications and application types that constitute the smart grid, the consensus definition more accurately reflects that the smart grid is a collection of applications whose individual definition and order of importance vary from utility to utility. As such, the industry will have no formal, strict and technical definition of the smart grid.

In spite of this uncertainty, SCE's approach brings determinism and engineering rigor to a domain fraught with unfounded ultimatums and speculative estimation. The process bounds and controls unknown issues, allowing them to fit into the equation, and be addressed by formal processes.

5.3.2 Symbiosis of Reusability and Modularity

The disciplined aspects of requirements engineering and risk management embedded in this approach allow the utility to develop a formal model representing the problem space. However SCE has taken this philosophy a significant step further by establishing reuse and modularity as priorities throughout the process.

Each of the components described herein is explicitly designed to facilitate reuse in future iterations. Additionally, the modular design aspects allow a holistic, all-encompassing, representative architectural model to be built step-by-step as the utility addresses individual problem spaces. These priorities prove to be highly complementary as the complete picture may be broken down, reorganized and reassembled while retaining the integrity of the model and forcing only minor re-validation of the linking equations.

5.3.3 Satisfaction of Interests

SCE was faced with a daunting problem in 2006: how to secure a system that had never been built before and was yet to be entirely defined. The Edison SmartConnect™ project was the beginning of an ambitious endeavor to transform the utility into the next generation of electric power service providers and implement what some were starting to call the “smart grid.” Yet no previous work on this problem existed and the utility had no markers to point the way.

Development of this approach has provided the necessary guidance and gone well beyond satisfying SCE’s interests and needs in the domain of complex systems security engineering. While no individual step is revolutionary, the overall assembly of processes has proven to be profound and compelling. SCE has illustrated a pioneering spirit and exemplary leadership in its security engineering approach, and endeavors to share this knowledge and experience with all those interested.

Contributors:

John Bubb, SCE

Jeff Gooding, SCE

Jeremy Mc Donald, SCE

Darren Highfill, Enernex