SECURITY PROFILE BLUEPRINT

Prepared for:

The SG Security Working Group (UCAIug)

&

The NIST Cyber Security Coordination Task Group

Prepared by:

The Advanced Security Acceleration Project for the Smart Grid (ASAP-SG)

Managed by:

EnerNex Corporation
620 Mabry Hood Road
Knoxville, TN 37923
USA
(865) 218-4600
www.enernex.com

Version 1.0
October 6, 2010

Revision History

Rev	Date	Summary	Marked
0.17	20091116	Numerous changes: • Added Revision History • Changed Utilisec references to SG Security • Miscellaneous revisions to Section 1 including deleting Section 1.1.1 • Major rework of Section 2 • Added content to Section 3	N
0.18	20091119	Clean version (no changes/comments visible) of v0.17	N
0.19	20091130	Misc changes	N
0.20	20091214	Added Figure 1 to document	N
0.21	20100927	Substantial updates to Sections 2.1 through 2.3	N
0.22	20100928	Updates to Sections 2.4 and 2.5. Minor changes based on comments on 2.1 – 2.3.	N
0.23	20101001	Update to Section 3; edits from comments in 2.1-2.4.	N
0.24	20101005	Substantial updates to front matter and Section 2.3.3. Revisions to Section 2 introduction. Edits throughout.	N
1.0	20101006	Complete version. Public release.	N

Executive Summary

The Smart Grid Security Profile Blueprint provides the electric utility industry along with supporting vendor communities and other stakeholders a framework, set of tools, and method to create and customize Smart Grid domain-specific security profiles. These security profiles specify security requirements that should be applied to the procurement, implemention, and configuration of Smart Grid systems. These requirements will ensure the high level of information assurance, availability and security necessary to operate a reliable system and maintain consumer confidence. The security profiles created by using this Blueprint will augment and clarify more general established standards and best practices for cyber security through the specification of usable, actionable, and traceable security requirements tailored to specific Smart Grid applications.

The primary audience of the Blueprint is any organization attempting to create a new security profile or customize an existing security profile; therefore the document is written for security architects from utilities, vendors, and system integrators that have experience with utility security. Other stakeholders, such as vendors, can use this document to understand how a particular set of security controls was selected as part of a particular security profile. The Blueprint is intended to produce requirements that are technology-specific but vendor-agnostic, and does this by defining a process for creating a security profile. This process includes the delineation of profile scope, creation of a logical reference architecture, definition of objectives for secure operation, performance of a failure analysis, recommendation of security controls, and validation of criteria for satisfaction of requirements.

Acknowledgements

The Advanced Security Acceleration Project – Smart Grid (ASAP-SG) would like to acknowledge the work of the primary authors, contributing authors, editors, reviewers, and supporting organizations. Specifically, we would like to thank:

- Participating utilities, including American Electric Power, BC Hydro, Con Edison, Consumers Energy, Florida Power & Light, National Grid, Oncor, and Southern California Edison
- Supporting organizations including The United States Department of Energy and the Electric Power Research Institute
- The utilities, vendors, consultants, national laboratories, higher education institutions, governmental entities, and other organizations that have actively contributed to and participated in the activities of the SG Security Working Group and the Usability Analysis Task Force.

The SG Security WG would also like to thank the Department of Homeland Security (DHS) Cyber Security Division, National Institute of Standards and Technology (NIST) Computer Security Division, North American Reliability Corporation (NERC) and The Common Criteria for the works that they have produced that served as reference material for the Security Profile Blueprint document.

The ASAP-SG Architecture Team included resources from Consumers Energy, EnerNex Corporation, InGuardians, Oak Ridge National Laboratory, the Software Engineering Institute at Carnegie Mellon University, and Southern California Edison.

Authors

Glenn Allgood

Len Bass

Bobby Brown

Kevin Brown

Jim Cebula

Slade Griffin

Teja Kuruganti

Howard Lipson

Jim Nutaro

Justin Searle

Brian Smith

James Stevens

Edited by: Darren Highfill and James Ivers

Table of Contents

1 Introduction

The term Smart Grid is defined in many ways by various stakeholders. Regardless of the specific definition, it represents the opportunity and challenge for utilities to update outdated infrastructure with a more intelligent power grid. To meet this challenge, utilities are starting to combine advancements in information technology with electricity infrastructure, enabling the transformation to a "smart" power grid or Smart Grid. The Smart Grid will use interconnected elements that optimize communications and control across the different segments of energy generation, distribution, and consumption. Near-real-time information allows utilities to manage the entire electricity system as an integrated framework, actively sensing and responding to changes in power demand, supply, costs, quality, and emissions across various locations and devices. Better information also enables consumers to manage energy use to meet their needs.

Recent funding appropriations in the American Recovery and Reinvestment Act have been earmarked toward stimulating increased deployment for smart grid technologies and applications. The media and policymakers, however, have expressed concerns regarding the potential risks and vulnerabilities of this enhanced power system. As the process of migrating to this type of infrastructure accelerates across the industry, it is important to understand the need for security and also define a set of effective security controls.

Features of the Smart Grid are intended to enhance the security of the national electrical infrastructure, but the extension of two-way digital communications could make protecting the power grid from a cyber attack a far more complicated mission – adding nodes to a network can introduce new openings for intruders. Rather than only trying to ensure the security of the current producers of bulk power in the USA, those responsible for Smart Grid security will have to account for potentially millions of new touch points

involving the end consumers and interconnections of systems that have traditionally been isolated.

Although mature for the business system environment, traditional security methods have been difficult to apply at the process-control level leading to weak, incompatible, and inconsistent implementations across the utility industry. To address this issue, the Smart Grid Security Profile Blueprint was created and then matured through its use to create several Smart Grid application-specific Security Profile documents.

The Smart Grid offers the ability to bring together complex, proprietary systems onto a common, standards-based network infrastructure will enhance communications, improve efficiency, help reduce costs, integrate renewable sources of energy, and promote more opportunities for innovation. Though the Smart Grid has a lot of potential to improve the way electric power is generated and delivered, the fact that this new paradigm also requires extra connectivity puts the power infrastructure at risk unless security strategies are evolved along with the grid.

A key element in the evolution of the Smart Grid is the convergence of the power grid, the communications infrastructure, and the supporting information infrastructure. Utilities and vendors alike are faced with a myriad of decisions when dealing with security and compliance requirements for these complex systems and simple or "one size fits all" security solutions are not adequate to address requirements at the various levels and domains.

While there are several high level or abstract standards and reference models available that can be used, there is an absence of tactical guidance on the subject to assist in the final implementations. Although effective when applied to systems within the business environment, these traditional security methods have been difficult to apply and tailor to utility process control systems leading to weak, incompatible, and inconsistent implementations across the utility industry. This deficiency is affecting the electric utility industry's ability to deploy appropriate cyber security measures within Smart Grid systems and components in a timeframe acceptable to both regulatory and legislative bodies.

To address this issue, the Smart Grid Security Profile Blueprint (heretofore “Blueprint”) and various Smart Grid application specific Security Profile (SP) documents have been created. These references provide guidance to refine security measures required throughout the various environments and domains within the Smart Grid. These documents focus on the security services that are important to secure the power grid, communications infrastructure and supporting information infrastructure.

The Blueprint and individual SP documents were developed to encourage and enhance security of smart grid systems and facilitate the broad adoption of consistent cyber security measures across the electric utility industry. These documents can be used individually for security system design or as part of the utilities overall security program.

1.1 *Purpose*

The purpose of the Smart Grid Security Profile Blueprint is to provide the utility industry along with supporting vendor communities and other stakeholders a framework, set of tools, and method to create and customize Smart Grid domain-specific security profiles. These security profiles identify security requirements that should be applied to Smart Grid system implementations to ensure the high level of information assurance, availability and security necessary to maintain a reliable system and consumer confidence.

The security profiles created by using this Blueprint will augment and clarify established standards and best practices involved in procuring, implementing, and operating Smart Grid applications. In short, the Blueprint and SPs are particularly relevant to some portions of a system's lifecycle. They are not, however, intended to cover every possible security need. For example, continuous risk management—an important concept to be managed throughout a system's lifespan—is not directly addressed by the Blueprint or any SP.

1.2 *Audience*

The primary audience of the Blueprint is any organization attempting to create a new security profile or customize an existing security profile. Other stakeholders, such as vendors, can use the Blueprint to understand how a particular set of security controls was selected as part of a particular security profile.

The Blueprint is primarily written for security architects from utilities, vendors, and system integrators, that have experience with utility security. The purpose of the Blueprint is to define a method for creating usable and actionable sets of security controls for particular smart grid applications (which are packaged as security profiles).

1.3 *Blueprint Objectives*

The Smart Grid Security Profile Blueprint provides an understandable and user-friendly framework, set of tools, and a method to create and customize smart grid domain-specific security profiles. This method includes the delineation of profile scope, creation of a logical reference architecture, definition of objectives for secure operation, performance of a failure analysis, recommendation of security controls, and validation of criteria for satisfaction of requirements.

The security profiles created by using this Blueprint will augment and clarify more general established standards and best practices for cyber security through the specification of usable, actionable, and traceable security requirements tailored to specific Smart Grid applications.

1.4 *Security Profiles*

The Security Profile documents provide prescriptive, actionable guidance for how to build-in and implement security for smart grid functionality. A Security Profile is a reference that identifies detailed security requirements for assets involved in supporting a specified collection of smart grid functionality. These specifications are agnostic to vendor and implementation. The potential users of a Security Profile are all stakeholders in the smart grid community who are concerned about security.

- Utilities that want to secure their systems.
- Vendors and service providers that want to offer products and services that satisfy established smart grid security requirements.
- Government agencies and consumers that want to know what is being done to secure smart grid resources.
- Standards development organizations that want to collect best practices for security in the smart grid.

Depending on the environment and/or the needs of the user, the Security Profile for an individual implementation may also be customized in various ways, such as:

- Identifying requirements that are not applicable given lack of support for specific use cases
- Changing strength of controls used to mitigate specific risks
- Deriving the set of requirements applicable to individual assets

In addition to identifying detailed requirements for specific smart grid functionality, the Security Profile provides traceability for these requirements from a designated set of use cases through a failure analysis process. Finally, the Security Profile provides validation criteria for satisfaction of requirements.

1.4.1 How is a Security Profile used?

1.4.1.1 Utilities

Utilities can use a security profile to better manage their relationships with vendors and government organizations with respect to security concerns and to increase internal security awareness.

A utility can use a security profile

- As a source of detailed security requirements for RFPs. Requirements from a security profile can be used “as-is” or customized to reflect organization-specific technology choices or priorities.
- As a resource that informs ongoing security/risk management activities.

- As a resource that informs trade-off decisions between security and other quality attribute requirements, such as reliability or usability.
- As a basis for auditing activities.
- As documentation of measures taken to address security concerns.

1.4.1.2 Vendors and Service Providers

Vendors and service providers can use a security profile to reliably identify security requirements applicable to their products or services.

A vendor or service provider can use a security profile

- As a source of applicable security requirements
- As a common frame of reference (or language) against which to target and identify their offerings
- As a benchmark to distinguish their offerings (as compliant)

1.4.1.3 Government Agencies

Government agencies can use a security profile

- As a resource in assessing measures being taken to secure the smart grid
- As a means to answer public concerns over what is being done to secure the smart grid
- As a basis for potential regulation or auditing activities

1.4.1.4 Consumers

Consumers can use a security profile

- As a resource in understanding measures being taken to secure the smart grid

1.4.1.5 Standards Development Organizations

Standards development organizations (SDOs) can use a security profile

- To identify needs not satisfied by existing standards
- As a mature draft or significant input into new standards development

2 Security Profile Creation Method

Creation of the security profile is a multi-stage process. It involves input from stakeholders and subject matter experts which is processed by security experts culminatng in creation of the security profile. Figure 2 - Security Profile Creation Process shows the method for creating a security profile, which involves the following steps:

1. Establish profile scope: The purpose of this step is to determine what application functionality will be considered in-scope for the security profile. Scope determination is achieved by soliciting formal input from subject matter experts and stakeholders; nominating functionality, system-level applications, and sub-system components for coverage; and proposing a set of use cases. These use cases are analyzed to understand the system components that are involved in implementing the functionality, and a subset of these components are determined to be in-scope. The security profile will recommend a set of baseline security controls for all in-scope components.
2. Define logical architecture: The purpose of this step is to document the logical architecture of the in-scope components to provide a context for security recommendations. Important information that is documented in this step includes the set of components, interfaces between in-scope components, interfaces between in-scope components and external components, and descriptions of the kinds of information and control signals passing over each interface. This step also documents a set of roles with definitions encompassing responsibilities and functionality for each role.
3. Gather security influences: The purpose of this step is to identify information that constrains or shapes security recommendations for in-scope components. This is

comprised of identifying non-functional characteristics that influence security recommendations (e.g.: ownership/control, physical access, communication links, and information sensitivity/longevity) and performing a detailed failure analysis of the use cases. The failure analysis is performed to determine potential failures within the system and components in-scope, as well as to identify the consequences of those failures.

4. Recommend security controls: The purpose of this step is to select the baseline set of security controls for in-scope components. Control selection is based on the outputs of previous steps, careful study of industry and government best practices and recommendations, and tailoring of appropriate controls to the needs of smart grid applications in-scope. All recommended security controls must be clearly allocated to one or more components and for each component the control must address at least one of its potential failures.

5. Validate profile: The purpose of this step is to validate security recommendations with a broader audience than the profile's author team. Validation is required to confirm the selected controls are appropriate, justified, and expectations of the stakeholders are met. All the controls are justified and appropriate. This may include some combination of activities like public review, directed review, and consultation with domain or technical experts.

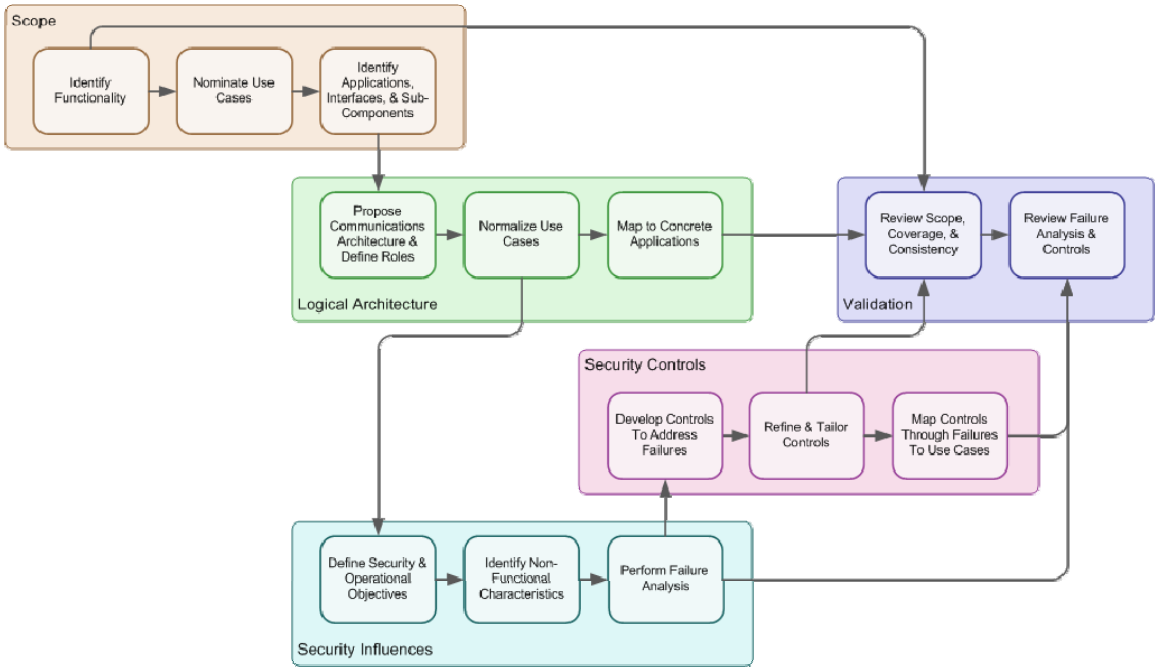

Figure 1 - Security Profile Creation Process

Each step is elaborated in the following subsections. The Security Profile for Distribution Management [ASAP-SG] was created using this method, and its contents contain good examples of the outputs each step documented here.

2.1 *Establish profile scope*

The first step in creating a security profile is to establish the scope of the profile. The scope defines which components and functionality will be considered in the development of security recommendations. Clearly identifying a scope allows subsequent activities to be pursued in the context of specific system information, which in turn allows more specific recommendations to be made.

Determining the scope of a profile is a judgment call. Some advice to keep in mind:

- maintain coherence of the security profile by including only related functionality (e.g., similar business functions that are implemented by a common set of components)
- include as complete a portion of related functionality as is feasible (e.g., if you include functionality related to steady state operations, also include functionality related to establishing and terminating normal operations)
- define a consistent boundary with respect to inclusion of components (e.g., if security controls are included for a component as used in one use case, security controls should be included for that component across all in-scope use cases)
- build as much as possible on mature, well-defined material that has been subject to widespread review and feedback

Regardless of the criteria for deciding what is in or out of scope, the results should be clearly documented in the security profile.

2.1.1 Inputs

Potential functionality can be documented in several ways, use cases and scenarios being among the more common. Community collections of potential use cases are available through efforts such as those documented on the SmartGridiPedia and NIST web pages. Profiles for emerging areas, where existing use cases are limited or non-existent, may require use case or scenario development in order to complete this step.

Use cases, like those found on SmartGridiPedia, are sometimes elaborated through detailed scenarios. Some scenarios on SmartGridiPedia define workflows between system components and actors that are very helpful in identifying the components that are needed to implement the identified functionality.

Regardless of the availability of use cases, stakeholder interviews are essential input for scope definition. Stakeholders can provide important insights regarding how functionality is grouped, location and purpose of system interfaces (common boundaries), and real-world interpretation of roles, interfaces, and other elements appearing in abstract reference material.

2.1.2 Process

As there is no right or wrong way to decide scope, the process for this step is rather general. Be sure to use a process that solicits buy-in from those expected to use the resulting security profile.

1. Determine the scope of functionality to be included in the security profile. This is usually a consensus exercise, moving from a general notion of the subject of a profile to a more precisely defined scope.
   - Write a general statement of the scope of the profile.
   - Identify or create a preliminary set of use case titles that define the application functionality under consideration. The use cases do not have to contain detailed steps and descriptions at this point, but should imply specific functionality.
2. Determine which system-level applications are in-scope. System applications should be defined in terms of most common industry usage while avoiding explicit brand/product references.
   - Identify and define each system application that will be covered. Definitions should indicate what functionality is included and what is not – especially for any potentially confusing or controversial terms.
   - Identify neighboring applications/systems and their points of interface.
3. Determine which system components are in-scope. Security recommendations will be created for each in-scope component as part of profile creation.
   - Determine the set of actors and components needed to realize each in-scope use case (this information will be revisited in the next step).
   - Identify the sub-set of components for which security recommendations will be generated—these are the in-scope components. Note that the application specific security profiles focus on requirements that can be imposed on system elements, rather than individuals or organizations.
4. Present a draft/proposed scope to funders and/or key stakeholders.
   - Indicate that the scope is proposed (as opposed to final) and invite feedback and critique by a specified date. A meeting or conference call after a brief period for review is recommended.
   - Modifications, if needed, may be purely semantic or technically fundamental.
   - Ensure that all parties are ultimately in agreement on the scope definition and impact on resources and schedule.

2.1.3 Outputs

The outputs of this step is a written summary of the scope of the security profile. This includes:

- The documented set of components that will be subjects of security recommendations.
- A description of all logical and physical boundaries explicitly indicating which components are in-scope and which are out.

- The documented set of applications that will be covered by the security recommendations, including real-world examples.
- Any explicit exclusions along with a brief explanation for each exclusion.

2.2 *Define logical architecture*

Once initial scope decisions have been made, the system must be further analyzed to understand the assets to be protected. This analysis first depends on establishing a perspective of the system that provides insights into how components communicate and interact.We refer to this perspective as a "logical architecture."

Two of the essential aspects of an effective and useful logical architecture are its ability to represent multiple different real-world instantiations and the degree to which system responsibilities and functionality are separated into cohesive groups under meaningful labels. We therefore use the concept of an abstract "role" to represent a distinct and cohesive set of responsibilities and functionality. In use case (or UML) parlance, a role has similarities to an actor but allows for more flexibility when mapping to components or products in real-world examples. Sometimes a single component or product may actually satisfy multiple roles, while in a different environment multiple components or products may need to be aggregated to satisfy the functionality of a single role.

The logical architecture must also effectively depict the relationships, interactions, and interdependencies between logical components and their interactions. This information is realized through an iterative detailing and refinement of the use cases initially identified in the scope definition phase. The original use case titles will frequently undergo significant modification, decomposition, and re-grouping as the embodied functionality is broken down and associated with logical architecture roles. Notably we have found it important that the use cases focus on the business processes that need to be secured, but explicitly do not describe any security functionality so as not to inadvertently engineer any assumptions about security into the model. All security functionality must be specified through requirements (discussed later).

A logical architectural view does not attempt to capture deployment information, such as allocation of functionality to hosts or network segments. Nor does it attempt to capture all terminology and physical configurations represented by different products.

In particular, you should capture the topology of components that shows

- interactions between in-scope components (internal interfaces)
- interactions between in-scope components and out-of-scope components or actors (external interfaces)
- the nature of each interaction, such as the type of information or commands flowing between components
- how long information resides at different points in the system

This information will be needed to understand the different needs, importance, and vulnerability of different components in the system. It will be used in subsequent steps to provide tailored security recommendations for the components defined in the logical architecture.

Defining a logical architecture is a difficult task, as one must avoid biasing the architecture towards any particular technology or product. Ideally, a reference architecture will be abstract enough (hence the "logical" part) to correctly represent a broad array of potential implementations, but be specific enough to allow strong security recommendations to be made.

In some cases, it may be necessary to identify one or more variant logical architectures that represent profound variations with significantly different security implications.

2.2.1 Inputs

Sometimes a logical architecture may already exist that can serve as the starting point for discussions about roles and use cases. Community collections such as those documented on the SmartGridiPedia and NIST web pages are again a good source. However, the process of identifying and refining roles and use cases may dictate that the logical architecture be significantly modified or possibly a new one built.

Regardless of the starting point, building a logical architecture depends on the ability to collect information about how components will interact to implement the application functionality.

Potential sources of this kind of information include

- detailed scenarios identifying components, actors, and workflows
- a collection of reference architectures characterizing known or planned implementations
- interviews with domain experts

2.2.2 Process

Again, there no one right way to define a logical architecture, and the steps followed will differ based on the kind of information that is available. This section documents one potential process, though others that generate the necessary outputs would be acceptable. The documented process assumes that good, detailed scenarios corresponding to the in-scope use cases are available and are the primary sources of information.

1. Aggregate information from source information (e.g., detailed scenarios). Information may come from different sources, and so may need to be normalized (i.e., unify terminology like component names). In particular, different sources may have inconsistent information flows (e.g., different paths for routing meter data to third-parties). Investigate such discrepancies to see if they represent viable, probable alternatives. If so, document the variation; if not, impose a consistent information flow in the architecture.

2. Re-examine component scope decisions. Now that a more complete picture of interactions between components is documented, determine if you have left out an important element of systems implementing this profile or if you've included a component that has little to do with the profile. Revise your component scope as needed.
3. Define all in-scope components. Because different organizations or products may use different words to refer to the same logical functionality, the profile must be clear about how it defines each component. Wherever possible, remain consistent with other widely used community sources (e.g., the NISTIR).
4. Describe all internal and external interfaces. This provides additional clarification as to the function of each in-scope component and necessary information for making security recommendations. Both internal and external interfaces should be described, as both present attack surfaces to be considered.
5. Define an initial set of roles in terms of responsibility and functionality, and propose relationships between the roles. This is the start of the actual logical architecture, however subsequent iteration and discovery of use case material are likely to foment substantial changes.
6. Iteratively refine the use cases identified in the scope. Break the use cases up into discrete chunks of functionality that are as small as practical – this allows focus on the purpose of the use case and will likely allow for some amount of modularity and re-use. Use cases should be oriented to roles (i.e. the “swim lanes” should have 1:1 correspondance to the roles).
7. Re-factor the roles, relationships, and use cases until a cohesive picture emerges. Use cases should all be written at the same level of detail.

2.2.3 Outputs

The output of this step is a logical architecture that defines the following information:

- the documented set of roles with definitions encompassing responsibilities and functionality for each role
- the logical topology of a system implementing the profile
- the documented set of use cases describing the functionality for which security recommendations will be generated
- a definition of each in-scope component in the logical architecture
- a description of each internal interface (i.e., descriptions of the kind of information and commands passing between in-scope components)
- a description of each external interface (i.e., descriptions of the kind of information and commands passing between in-scope components and external components or actors)
- a description of how long important information resides at each in-scope component

2.3 *Gather security influences*

Where the logical architecture focuses on the structure and function of a smart grid application, this step focuses on non-functional characteristics that also influence security recommendations. Such characteristics include

- security and operational objectives (i.e., broad statements of desired level of security and ideal operational behavior)
- business, regulatory, organizational, and technological constraints (i.e., various factors that limit or mandate the security controls that can be put in place, such as available bandwidth)
- a failure analysis indicating different points and types of failures that can arise in the logical architecture
- other non-functional descriptions that influence security decisions (e.g., delineating asset ownership or deployment location and resulting physical access)

As with the logical architecture, these influences may need to be described more abstractly in a security profile than in an application of a profile to a particular system. This is because the context refines many of these characteristics (e.g., by limiting some technical options in order to remain compatible with legacy systems or defining specific risk tolerances). As such, a formal risk analysis is not typically performed when creating a security profile meant to apply across a broad range of deployed systems. Construction of such an analysis would be a good way to refine an existing profile as part of an application in the development or procurement of a system conforming to the profile.

The end goal of this step is to generate any other information (in addition to the logical architecture) that is needed to make justified security recommendations. This could include different characteristics for different profiles, however, the following are recommended as a set of actions that should be taken for all security profiles (in this order)

1. Define security and operational objectives.
2. Identify any non-functional characteristics pertinent to security recommendations that are not defined in the logical architecture.
3. Perform a failure analysis.

If business, regulatory, operational, or technological constraints that are common across systems that are the subject of a profile can be be identified, this should be done prior to the failure analysis. These actions are described in the following sub-sections.

2.3.1 Define security and operational objectives

2.3.1.1 Inputs

The information needed to identify security and operational objectives should come from a variety of sources, including

- stakeholder interviews and personal domain expertise
- public statements from utilities, vendors, and regulatory and government agencies regarding desired security goals

2.3.1.2 Process

The goal of a security profile is to establish a computing and communications environment in which the scoped system can successfully and securely operate. Meeting this goal requires that a number of security and operational objectives that support that goal are achieved. Objectives for the scoped system are identified and used throughout the failure analysis (see Section 2.3.3). These objectives served as the "ground rules" for the scoped system.

The process involved is largely one of brainstorming, refinement, and conversation among people well versed in the domain.

Remember that this step is not about defining precise security controls, which may be a natural inclination. Instead, focus should be on defining broad goals whose satisfaction define success. A workable set should fit on one page and not be predisposed to any particular security solution. Thinking in terms of broad confidentiality, integrity, and availability goals may be helpful.

2.3.1.3 Outputs

The output of this step is a concise set of broad security and operational objectives. These objectives will be used during the failure analysis and control selection.

2.3.2 Identify non-functional characteristics

2.3.2.1 Inputs

The information needed to complete this step comes largely from stakeholder interviews, personal domain expertise, and research.

2.3.2.2 Process

This step is less prescriptive than most as it may address different characteristics in different profiles. The general process is as follows

1. Determine which non-functional characterstics are relevant to the security profile under development.[1] Such characteristics often differentiate among roles in the logical architecture in a way that indicates different security needs (e.g., by noting that some are physically protected while others are in publicly accessible areas).

[1] The line between non-functional characteristics and various constraints can be blurry. Is a description of asset ownership the description of a characteristic or a constraint? It doesn't really matter. What matters is that the appropriate information is identified so that it can be used in following steps.

Some characteristics to consider include

- ownership and control of systems implementing a role
- physical access to deployments of a role (protected or not)
- nature of communication links (e.g., wired or wireless, bandwidth limits, and exclusive or shared use)
- information sensitivity (e.g., control signals, PII, or publicly available information like price signals)
- information longevity (e.g., does information merely pass through a role or is it stored for some period of time)

2. For each such characteristic, record some description for each role in the logical architecture relative to the characteristic. That can be done in several ways, such as in tables (e.g., noting likely bandwidth constraints on different links) or in graphical overlays (e.g., noting which roles are owned by utilities vs. which are not).

2.3.2.3 Outputs

The output is a description of each identified characteristic for each role in the logical architecture. The organization of this information can vary (e.g., tabular, graphical, or text).

2.3.3 Perform a failure analysis

The purpose of a failure analysis is to determine potential failure points within a system and the consequences of those failures. Failure analysis for natural or innate reliability is an established and widely practiced discipline. The FMEA (Failure Modes and Effects Analysis) and FMECA (Failure Modes, Effects, and Criticality Analysis) approaches are classical and well know methods in this field (see, e.g., http://en.wikipedia.org/wiki/Failure_mode_and_effects_analysis) that lead to risk anlaysis and assessment in the form of risk registers and consequence management profiles.

Where a FMEA is qualitative in nature allowing a subjective assessment of a Risk Priority Number (RPN), a FMECA is quantitative requiring data to support a calculated criticality measure that is based on derived failure rates. Therefore, including the FMEA and FMECA, two categories (subjective vs objective) and three analysis venues can be considered for conducting a cyber-security failure analysis (see the following Table).

Table 1 - Cyber-Security Venues for Conducting System Failure Analysis

Analysis Technique	Objective	Subjective
FMECA	X	
FMEA		X
Blueprint		X

In the objective category, assuming sufficient information concerning failure rates, probability distributions, and a clear perspective on the information architecture, the FMECA would be the choice. If evidence doesn't exist to support informative conclusions about failure rates and only an abstract view of the architecture exists, then either the FMEA or the process outlined below would be the choices.

2.3.3.1 Inputs

This step makes use of most of the information that has been gathered in the creation of a profile to this point. Specifically

- roles are defined in terms of their intended function
- the logical architecture shows the possible interactions among roles
- use cases describe how roles interact to perform their functions
- security and operational objectives describe the system's goals
- non-functional characteristics provide additional information that could point to potential failures

2.3.3.2 Process

The purpose of this simplified failure analysis is to enumerate the types of role and system-level failures that will be addressed by the security profile. This is facilitated by the following appoach:

1. Iterate through each step of each use case. For each step, identify the primary role (the swimlane within which that step resides) and ways in which that role could fail to satisfy its operational objectives within the scope of the use case. The list of ways in which the role could fail to satisfy its operational objectives is the "list of failures" for that role.
2. After a first pass is made through all use cases, consolidate the failure list to remove duplication and minor variants (through generalization).
3. Make another pass through the use cases, using the consolidated list of failures. In this pass, examine each step to see if any failures were missed and replace the

original failures with the equivalent failure from the consolidate list. This will result in a unified set of failures across all roles, simplifying control selection.

4. Add to the consolidated list of failures any new failures (component- or system-level) that may be induced by a listed failure (e.g., failure to send induces a failure to receive).
5. Eliminate from the consolidated list any failures that are deemed out of scope and record the reason for its exclusion.

2.3.3.3 Outputs

The outputs are a table (or set of related tables) that

- lists all identified failures (this is useful as a separate table)
- maps each use case step to a role and the set of all failures that apply to the step; each failure should also be mapped to the security and operational objective or core function that it jeopardizes (this table provides fine-grained traceability from roles and failures to the use case steps in which the failure could occur)
- maps each role to all failures that apply to it (a summary table, by role; this table provides a course-grained traceability between roles and the failures they could exhibit or encounter)

2.4 *Recommend security controls*

Upon completion of the failure analysis, security controls can be assigned to all in-scope components. Minimally, a security profile must define a set of *baseline* security controls. Optionally, additional controls can be defined as enhancements.

All security controls must be clearly allocated to one or more components and for each component the control must address at least one of its potential failures. Control selection and definition is more art than science, and requires good knowledge of the application area, relevant technologies, and security principles.

For controls to provide effective guidance, they must be actionable. Finding the correct level of specificity to make controls actionable can be challenging at times, however the objective should be to get as technically specific as possible without prescribing sub-system design or identifying specific products or vendor names. For example, specifying acceptable algorithms and key sizes is appropriate, as is delineating requirements for handling of key material within a device; however dictating chip architecture is likely to inhibit market and solution development without necessarily solving a specific problem.

Controls specified in a security profile should also be limited to those controls that are directly relevant to securing the system under discussion. General security good practice requirements are unhelpful, as they may already be found in any number of respected works and tend to dilute the focus on controls that are specific to the environment and system at hand. Therefore the most effective controls are written with a specific system in

mind, using a broadly recognized and accepted catalog (e.g. NIST 800-53) as both inspiration and a means to ensure adequate coverage.

2.4.1 Inputs

Inputs to this step include

- the documented logical architecture
- identification of which components are in-scope
- documentation of any additional security-related constraints
- a list of operational security objectives
- a normalized list of potential failures for each component
- reference material documenting industry and government best practices and security controls

2.4.2 Process

Security control selection and development is largely based on expertise and good judgment. A sketch of one reasonable approach follows.

1. Starting with the normalized list of potential failures from 2.3.3 above, develop a set of controls that address each potential failure.
    a. Using a reference set of security controls from a related domain is effective at providing inspiration, however controls should be prescriptive and actionable to a level that is helpful for both asset owners and system developers. Good reference sets include those from DHS [Catalog of Control Systems Security: Recommendations for Standards Developers], NIST [NISTIR 7628], and the Common Criteria [v3.1].
    b. Keep a record of each referenced control with the new controls written. This helps with ensuring coverage later.
2. Map the controls through the failures to the use case steps and roles.
    a. Some controls may fully address potential failures by applying them at specific steps in the use case. In this case, the controls may be most appropriately invoked at the time of system configuration or operation. Other controls may apply to all use case steps for a specific role, whereupon they should be specified as bound to the role and all components that satisfy that role.
    b. Consider each in-scope component in turn. Based on a component's role in the logical architecture and its security-related characteristics, determine whether the candidate security controls are appropriate. Note that a particular component may be incapable of supporting a necessary security control (e.g., because of insufficient memory or processing power). In that

case, a security control may be allocated elsewhere in the system to provide the necessary security properties.

3. Refine each control as needed to suit the unique needs of the smart grid.
4. After selecting controls for each component, review the collection, looking for gaps or weaknesses in the allocation of controls across the system. Make any necessary adjustments to ensure that the desired security properties hold not only for the components, but also for the system as a whole. The selected controls must collectively address all failures associated with all components.

2.4.3 Outputs

The output of this step is a set of security controls for each in-scope component, applied at either the system design and build level or at the time of instantiation and configuration, and justification of each control in terms of the failures that it addresses for its targeted components.

2.5 *Validate profile*

This section describes a variety of validation activities that should be performed prior to completing a security profile. Ideally, rather than leaving all of the validation activities for the end of the process, many of these activities should be performed throughout the process in order to provide feedback as early as possible.

Validation can be done a number of different ways, but should address issues such as whether the recommendations are complete, adequate, appropriate, justifiable, and usable. Validation can be conducted in a variety of ways. Some possibilities include

- directed reviews focused on particular issues
- requests for comments from particular domain or technical experts
- an open call for comments from volunteers
- a formal voting process

Regardless of the specific process used, it is important to get independent feedback from qualified personnel who were not involved in the profile's creation. This section focuses on the issues to be validated rather than the techniques used to conduct the validation.

2.5.1 Inputs

The primary input to this step is a draft security profile. Additional inputs, such as qualified reviewers and stakeholder expectations, are also necessary.

2.5.2 Process

Validation activities should strive to confirm that a security profile meets the needs and expectations of its stakeholders. There are many small steps that contribute to such an outcome, including the following recommended steps

1. Review profile scope with key stakeholders (e.g., domain experts and funders) to ensure coverage of the target smart grid application area. Confirm that security recommedations for the components within the scope will be sufficient for stakeholder needs.
2. Review the logical architecture with key stakeholders. Confirm that the abstract use cases, roles, and their relationships address current and anticipated concrete architectures. Use the mappings between the logical architecture and concrete architectures as a tool in this review.
3. Confirm that all components and functions identified in the scope are addressed by some element of the logical architecture (i.e., nothing in scope was omitted). Confirm that the logical architecture does not include elements or functions that are not in scope.
4. Confirm that the use cases and logical architecture are consistent. There should be an interaction between elements of the logical architecture if and only if the elements interact in at least one use case.
5. Review the logical architecture for consistency with other smart grid efforts such as the NISTIR 7628 (http://csrc.nist.gov/publications/PubsNISTIRs.html#NIST-IR-7628) and the SGIP SGAC (http://collaborate.nist.gov/twiki-sggrid/bin/view/SmartGrid/SmartGridArchitectureCommittee). Wherever possible, consistent terminology should be used. Differences should be noted.
6. Review security and operational objectives for the target smart grid application area with key stakeholders. Confirm that these objectives are adequately covered by the high-level specifications of security goals and use case functionality within the scope of the profile.
7. Review the non-functional characteristics (e.g., requirements for safety, security, privacy, performance. latency) of the target smart grid application area with key stakeholders. Confirm what is currently addressed in the security profile and look for anything that is missing.
8. Review the failure analysis for completeness. Confirm that all use case steps and all roles have been considered. Determine if there are there failures noted in one case that should also be noted in other cases. Failures should be separately noted for each step of each use case (and with the associated role).
9. Review the failure analysis for justification. Confirm that all failures that are noted could jeopardize the satisfaction of the security and operational objectives or the core function of the system; this tracability should be recorded.

10. Review the failure analysis with key stakeholders. Confirm the traceability to objectives and determine if any key failures are missing.
11. Review the security controls for completeness. Confirm that a control that addresses a failure has been assigned to all roles mapped to that failure. Review a categorization of the recommended controls by purpose (e.g., deterrence, protection, detection, reaction, and recovery) and confirm that sufficient controls of each type are defined and appropriately assigned across roles.
12. Review the security controls for adequacy. Confirm that the combined set of controls assigned to a role is sufficient to address the failures mapped to that role (this includes consideration of the role's position within the architecture). Confirm that the combined set is sufficient to satisfy the security and operational objectives.
13. Review the security controls for appropriateness. Confirm that each control assigned to a role is reasonable for that role (e.g., that there are sufficient resources or that the control is not unduly expensive). Determine if an alternative control (that is preferable in some way) would provide equivalent security.
14. Review the security controls for justification. Confirm that each control is justified by explicit identification of the failure(s) that it addresses. Confirm whenever a control is assigned to a role, that the role is mapped to a failure that is addressed by that control.
15. Review the security controls for usability. Confirm that key stakeholders (e.g., vendors, utilities, integrators) are able to quickly identify the controls most relevant to a given need (e.g., a specific product). Confirm that each control is clearly written, avoids specific vendors or products, and is concrete enough to be actionable.
16. Review the security controls with key stakeholders. Confirm that the recommended controls are reasonable and sufficient to meet stakeholder needs.

2.5.3 Outputs

The output of this step is a validated security profile that is suitable for publication and use by a variety of stakeholders. Any issues raised during validation should be addressed by revisions incorporated in the final version of the security profile. The feedback and revision process should not be documented in the profile (aside from a change log), only the final result.

3 Security Profile Customization

The individual Security Profile documents represent minimum baselines of security controls for their respective smart grid applications. These security controls are a combination of controls that must be built into each system or device by their respective manufacturer and controls that must be deployed and maintained by the utiltity purchasing the device or system. A utility or vendor may have a need to modify these controls based on their intended use of the security profile, the identification of additional risks or limitations associated with a specific system architecture, or the adaptation to an organiztion's unique environment or requirements. In this case, a customized or "tailored" security profile may be warranted. The tailoring effort for a Security Profile typically includes one or more of the following:

- the identification of the applicable controls needed for an organization's intended use of the security profile
- the application of appropriate scoping guidance;
- the specification of compensating security controls, if needed;
- the specification of organizationally defined parameters where required by the security control.

Revised or changed security controls from the original baseline set should be carefully documented to show not only the changes and revisions to the control, but also the rationale or basis for the changes.

3.1 *Intended Use of a Security Profile*

Security profiles are designed to be used in whole or in-part by several different parties. For instance, vendors may use security profiles to incorporate security controls needed for the development of products and solutions. On the other hand, electric utility companies may use security profiles to achieve security objectives through activities such as:

- developing security requirements for procurement activities
- deploying, configuring and managing a system
- evaluating planned or deployed architectures

In some cases, an organizations intended use of a security profile may not require the full set of prescribed security controls. For instance, vendors may determine that some security controls apply only to a utility's deployment or mangment stages of a product life cycle and does not affect the featreset or design of their device. In turn, utitilies using a security profile to guide their secure deployment of a purchased product may identify several security controls that have been built into their products and are not applicable to their deplomentand management activities. Reguardless of the situation, careful consideration should be given to each control when determining its applicability to the organizations task at hand, and each disguarded control should be carefully documented.

3.2 *Scoping Guidance*

Scoping guidance provides a utility with specific terms and conditions on the applicability of each baseline security control to the utility's specific system and implementation. The issues that follow may affect the manner in which some of the baseline controls outlined in the individual Security Profiles apply to the system:

3.2.1 Technology-related issues

- Specific technologies - Security Controls that address specific technologies are only applicable to systems that use those technologies.
- A particular component may be incapable of supporting a necessary security control (e.g., because of insufficient memory or processing power). In this case, a security control may be allocated elsewhere in the system to provide the necessary security properties. (See Section 3.2.)
- Software, firmware or hardware that do not already exist or are not available in COTS products do not require that they be developed. Where the processes are not available or technically feasible, compensating security controls, implemented through non-automated mechanisms or procedures, may be used to satisfy the requirement. (See Section 3.2.)

3.2.2 Common security controls (Network)

Security controls provided as common controls at the network level may not need to be duplicated at the system/application level. This scenario may occur when a utlity network, or portion thereof, supports more than a single smart grid application. (e.g., the utility WAN may support both AMI and Distibution Automation applications. In this case, a security control deployed at the network level would satisfy the security control requirement for both systems rather than being duplicated for each.)

Every control in a security profile baseline must be addressed either through common security controls or by the system component.

3.2.3 Infrastructure issues

Security controls that refer to organizational facilities are applicable only to those portions of the facilities that directly provide support for the information system.

3.2.4 Scalability issues

Security controls must be scalable to the size and complexity of the system implementing the controls and the system's level of impact to ensure a cost-effective, risk-based approach.

3.2.5 Risk-issues

Security controls can be modified and/or enhanced based on a risk assessment that is specific to the utility's system and implementation. Utilities modifying security profiles should consider the affects of the following risks and impacts to the devices, components, and systems in their specific architecture.

- the permanent or temporary loss of any portion of any system
- the compromise (loss of control) of any portion of any system
- the public leakage of data being accepted, transmitted, or stored on any portion of any system
- the modification of data being transmitted or stored on any portion of any system
- the social engineering of humans with access to any system

When considering risks posed by malicious attackers, the utility should identify every device or communication channel accessible to potential insiders, both insider and outside the organization controlling the system. All physical and logical system interfaces should be considered potential attack vectors, as these human and machine interfaces are primary targets for attackers.

By completion of this process, the utility should create a list of applicable risks and impacts for the systems they are considering.

3.3 *Compensating Controls*

Compensating security controls are the management, operational, or technical controls employed by an asset owner in lieu of any prescribed baseline control outlined in the individual Security Profile documents. These compensating controls should provide equivalent or comparable protection for the smart grid system or component and their use should be tied to a documented and sound rationale and justification including risk assessment and acceptance.

The profile development approach guides the reader through the process developed by the ASAP-SG team for determining controls required for given failures (impacts) for roles and the functionality they implement (use cases), thereby providing traceability and justification for each of the controls selected. By utilizing this tracability built into each security profile, organizations may determine the justification for each control and identify the core risks that must be address by any compensating controls.

The following steps are recommended for the proper selection of compensating controls:

1. Identify the security control that can not met or implemented for a specific system or device
2. Identify the applicable roles for the specified system or device
3. Identify the use cases and failure modes that apply to the identified role
4. Determine which failure modes in each use case step are mitigated by the existing control that cannot be met or implemented
5. Determine suitable compensating controls that address all identified failures modes that were mitigated by the original control

3.4 *Organizationally Defined Control Parameters*

Security controls that contain organizationally definable parameters (i.e., assignment or selection operations) provide a capability for the system owner to define the controls' parameters to comply with organizational policy. Controls with definable parameters must be reviewed and the appropriate value assigned to the parameter. Applicable laws, executive orders, directives, policies, standards, instructions, regulations, or procedures may invoke values that are more restrictive. Additionally, risk assessments may indicate the need for the system owner to set a control's parameters to more restrictive values.

4 References

- Electric Power Research Institute (EPRI). 2009, June. Report to NIST on the Smart Grid Interoperability Standards Roadmap.
- National Institute of Standards and Technology. 2010, August. NISTIR 7628 – Guidelines for Smart Grid Cyber Security: Vol. 1, Smart Grid Cyber Security Strategy, Architecture, and High-Level Requirements; Vol. 2, Privacy and the Smart Grid; Vol. 3, Supportive Analyses and References. http://csrc.nist.gov/publications/PubsNISTIRs.html#NIST-IR-7628
- Department of Homeland Security, National Cyber Security Division. 2009, September. Catalog of Control Systems Security: Recommendations for Standards Developers.
- National Institute of Standards and Technology. 2007, December. NIST SP 800-18 Rev. 1 – Guide for Developing Security Plans for Federal Information Systems.
- National Institute of Standards and Technology. 2007, December. NIST SP 800-39 (second public draft) – Managing Risk from Information Systems.
- National Institute of Standards and Technology. 2007, December. NIST SP 800-53 Rev. 2 - Recommended Security Controls for Federal Information Systems.
- National Institute of Standards and Technology. 2007, September 28. NIST SP 800-82 - Guide to Industrial Control Systems (ICS) Security (2nd DRAFT).

- The Common Criteria. 2007, September. Common Criteria v3.1 – Part 2: Security Functional Requirements Release 2. The Common Criteria. Retrieved from
- The Common Criteria. 2007, September. Common Criteria v3.1 – Part 3: Security Assurance Requirements Release 2. The Common Criteria. Retrieved from
- SmartGridiPedia – www.smartgridipedia.org
- UCA International Users Group – SG Security Working Group. 2009, October. Security Profile for Advanced Metering Infrastructure (Draft 0.49).
- Advanced Security Acceleration Project for the Smart Grid. 2010, August. Security Profile for Distribution Management (Draft 0.12).