System Security Requirements Document

Version 0.3 Draft

Abstract

Advanced metering infrastructure systems promise to deliver support for dynamic pricing models, and to improve both the stability and reliability of the electric grid, but with a greater need for strong security throughout the architecture. In this paper, we identify the security threats to be considered in advanced metering systems. Additionally, we use qualitative metrics to rank the threats so that mitigations can be applied both effectively and efficiently. Finally, in the appendix, we present an extended set of common criteria threat material for inclusion into an advanced metering system level protection profile.

Advanced Metering Infrastructure systems offer a tremendous amount of potential, yet they introduce the requirements for industry proven, strong, robust, scalable, and open standards-based security. The goal of this working group is to define an exhaustive list of the potential security threats to the systems, and to perform detailed analysis of each threat to determine the threat levels and risks that it presents.

Revision History

Date	Version	Description	Author
<dd/mmm/yy>	<x.x>	<details>	<name>

Table of Contents

1 Introduction

(Update)
Advanced Metering Infrastructure (AMI) is a transforming technology that has broad impact on the energy market and its consumers. AMI allows utilities to balance supply, demand, and capacity making a smarter, more efficient, grid by pushing aspects of grid monitoring and control out to the endpoints of delivery. Stakeholders are implementing the systems and technologies required to deploy AMI today.

Advanced metering infrastructure systems promise to provide advanced energy monitoring and recording, sophisticated tariff/rate program data collection, and load management command and control capabilities. Additionally, these powerful mechanisms will enable consumers to better manage their energy usage, and allowing the grid to be run more efficiently from both a cost and energy deliver perspective. These advanced capabilities will also allow utilities to provision and configure the advanced meters in the field, offering new rate programs, and energy monitoring and control. With the advanced functionality, however, comes great responsibility. It is the purpose of the Advanced Metering Infrastructure Security Task Force (AMI-SEC) to provide utilities with sufficient guidance to build security into the basic fabric of this deployment.
In this document, we develop a qualitative methodology for identifying key AMI assets, their threats, vulnerabilities, and risks to support security control development. While many such methods exist for information technology and industrial control systems today, no method is adapted for the needs presented by the increased exposure of the AMI field systems. The method used proceeds by characterizing critical assets and their security concerns, system threats, critical asset vulnerability, and concludes with a method for analyzing risk. We next apply the method to a representative high level set of AMI assets.

This Security Risk Assessment (SRA) is a tool to help stakeholders identify the risk values in each AMI security domain, and in turn make effective decisions about how to mitigate those risks.

1.1 Purpose

(Update)

The objective of this System Requirements Document (SRD) is to provide an authoritative source of functional and non-functional system requirements for use by both Utilities and Vendors who procure, design and develop security related functionality for AMI systems.

1.2 Scope

(Update)
This document provides guidance for conducting the SRA in support of AMI architecture development. Organizations involved with AMI deployments will find this document to be a valuable resource in understanding AMI system risk. This assessment is designed to address the

specific security needs, organizational objectives, utility products and services, and processes and specific practices in regard to utility AMI deployment.
Security issues are elicited and aggregated for AMI critical assets from Premise Edge Services to Utility Operations. This assessment does not address non-AMI utility networks.
AMI-SEC has defined and tailored a risk assessment methodology specifically for AMI that includes:

- Identification of security domains,
- Identification of key AMI assets for each security domain,
- Description of security concerns for each asset,
- Identification of threats and threat agents,
- Evaluation of vulnerabilities associated with assets and security domains,
- Consideration of attack likelihood, and
- Evaluation of successful attack consequences.

The valuation of asset security concerns is considered input to the risk assessment methodology utilities may use to determine asset exposure and ultimately, control selection. This document does not advise mitigating measures or prescribe controls against risk determination. Control recommendations are conducted in a separate document.

1.3 Definitions, Acronyms, and Abbreviations

This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SSR. This information may be provided by reference to one or more appendixes in the SSR or by reference to other documents.

1.4 References

[BISHOP02] Bishop M.A. The Art and Science of Computer Security, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 2002
[CNSS4009] National Information Assurance (IA) Glossary, May 2003.
[JAQUITH07] Jaquith, A. Security Metrics: Replacing Fear, Uncertainty, and Doubt, Addison Wesley Professional Co., Inc., Boston, MA, 2007.
[LANDWEHR94] Landwehr C.E., A. R. Bull, J. P. McDermott, and W. S.Choi. “A taxonomy of computer program security flaws”. ACM Computing Surveys (CSUR), 26(3):211–254, September 1994.
[LEMAY07] LeMay M., G. Gross, C. Gunter, and S. Garg. "Unified Architecture for Large-Scale Attested Metering". HICSS, p. 115b, 40th Annual Hawaii International Conference on System Sciences (HICSS'07), 2007.
[NISTSP800-30] NIST SP 800-30 Risk Management Guide for Information Technology Systems, July 2002.
[NISTSP800-53] NIST SP 800-53 Rev. 2. Recommended Security Controls for Federal Information Systems. December 2007.
[NISTSP800-82] NIST SP 800-82 2nd Draft Special Publication 800-82, Guide to Industrial Control Systems (ICS) Security, 2007.
[NISTIR7298] NIST IR 7298. Glossary of key information security terms. April 25, 2006.
[OWASP] http://www.owasp.org/index.php/Category:Vulnerability

[PARKER02] Parker, D.P. “Toward a New Framework for Information Security”, The Computer Security Handbook 4th Edition., John Wiley & Sons, 2002.
[SALEH07] Saleh, M. S., Alrabiah, A., and Bakry, S. H. “Using ISO 17799: 2005 information security management: a STOPE view with six sigma approach”. Int. J. Netw. Manag. 17(1):85-97, January 2007.
[SHIREY00] Shirey R., "Internet Security Glossary", RFC 2828, May 2000.
[SPP05] System Protection Profile – Critical Infrastructure Process Control Systems, June 2005.

1.5 Overview

2 Overall description

2.1 System perspective

2.1.1 System Functions

This subsection of the SRS should provide a summary of the major functions that the system will perform. Sometimes the function summary that is necessary for this part can be taken directly from the section of the higher-level specification (if one exists) that allocates particular functions to the software product. Note that for the sake of clarity
a) The functions should be organized in a way that makes the list of functions understandable to the customer or to anyone else reading the document for the first time.
b) Textual or graphical methods can be used to show the different functions and their relationships. Such a diagram is not intended to show a design of a product, but simply shows the logical relationships among variables

2.1.2 System Interfaces

2.1.3 System Adaptation Requirements

This should a) Define the requirements for any data or initialization sequences that are specific to a given site, mission, or operational mode (e.g., grid values, safety limits, etc.); b) Specify the site or mission-related features that should be modified to adapt the software to a particular installation.

2.1.4 System Constraints

This subsection of the SRS should provide a general description of any other items that will limit the developer’s options. These include

2.2 Assumptions and dependencies

This subsection of the SRS should list each of the factors that affect the requirements stated in the SRS. These factors are not design constraints on the software but are, rather, any changes to them that can affect the requirements in the SRS. For example, an assumption may be that a specific operating system will be available on the hardware designated for the software product. If, in fact, the operating system is not available, the SRS would then have to change accordingly.

Topics:
a) Regulatory policies;
b) Hardware limitations (e.g., signal timing requirements);
c) Interfaces to other applications;
d) Parallel operation;
e) Audit functions;
f) Control functions;
g) Higher-order language requirements;
h) Signal handshake protocols (e.g., XON-XOFF, ACK-NACK);
i) Reliability requirements;
j) Criticality of the application;
k) Safety and security considerations.

(Review)
The following assumptions are listed to better clarify the scope of the risk assessment problem within the advanced metering infrastructure system [SPP05].
• AMI is a new application domain for system stakeholders, requiring new application of risk assessment, and subsequent security controls prescription.
• Consumers of this document have the ability to identify inputs to the risk assessment process.
• Consumers of this document are responsible for mapping and adapting its tenets to the protection of the value of their individual business values.
• An AMI system security design should incorporate principles of system survivability.
• Stakeholders for this document give preference to openness in security standards, guidelines, methodologies, and ultimately technology.

2.3 Requirements Assessment (Update this section)

2.3.1 Requirements Assessment Steps

There are many definitions of risk, but each has different implications for the nature of the AMI security problem. We leverage two definitions of risk that match the AMI community concerns

- A systems definition of **Risk***: The level of impact on organizational operations (including mission, functions, image, or reputation, organizational assets, or individuals resulting from the operation of an information system given the potential impact of a threat and the likelihood of that threat occurring.* [NIST800-53 Rev2]
- How to compute Qualitative **Risk***: a function of the likelihood of a given threat-source's exercising a particular potential vulnerability, and the resulting impact of that adverse event on the organization.* [NIST800-30]

We adapt a methodology of understanding AMI critical system asset risk. The risk assessment is presented in terms of a static assessment in this document, but must become part of a recurring risk management process for utilities implementing AMI-SEC recommendations to make it compliant with a goal of system survivability.
The following steps are taken directly from NIST 800-30 as a reasonable process for determining and documenting qualitative asset risk:

- Step 1 – System Characterization (Asset Identification for the purposes of AMI)
- Step 2 – Threat Identification

- Step 3 – Vulnerability Identification
- Step 4 – Control Analysis [not considered by this document]
- Step 5 – Likelihood Determination
- Step 6 – Impact Analysis
- Step 7 – Risk Determination
- Step 8 – Control Recommendations[not considered by this document]
- Step 9 – Results Documentation

For the purposes of the initial assessment, Steps 4 and 8 of the NIST SP 800-30 process are not addressed, but rather deferred to a future design document as this document presumes no specific system architecture. As an organization matures and systems are deployed, the utility can easily incorporate existing mitigations into their process. Note Steps 2, 3, 4 and 6 may be done in parallel after step 1 is completed. We will describe AMI specific policies for assessing risk in each of these steps below.

2.3.2 Mapping Risk through Security Domains

In the interest of approaching risk assessment in a way that is manageable, scalable and traceable, this document utilizes the IntelliGrid concept of Security Domains to aggregate logically cohesive system security requirements. A Security Domain (SD) represents a set of resources (e.g. network, computational, and physical) that is governed/secured and managed through a consistent set of security policies and processes. Thus each Security Domain that might be considered for AMI-SEC is responsible for its own general security process (e.g. Assessment, Policy, Deployment, Monitoring, and Training).
A Security Domain provides a well-known set of security functions that are used to secure transactions and information within that domain. We scale our risk assessment process by grouping AMI assets into Security Service Domains and subsequently treating risk by domain. This approach manages the explosion of relationships possible across the number of assets, threats, and vulnerabilities, and allows the mapping of Security Objectives (sometimes called Security Functional Requirements) to Security Service Domains. The rationale and design of the AMI security domains is given in a separate document.
Figure 1 - Risk Assessment Element Mapping illustrates relationships considered for mapping approach.

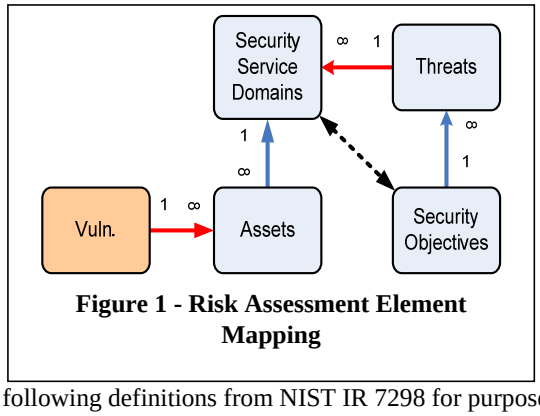

Figure 1 - Risk Assessment Element Mapping

AMI-SEC utilizes the following definitions from NIST IR 7298 for purposes of the mapping process:

Asset: *A major application, general support system, high impact program, physical plant, mission critical system, or a logically related group of systems.* (Note: this is a systems definition of the term "asset," which is appropriate for this level of analysis. Other uses of the term in this document are accompanied by explanation or definition.)

Threat: *Any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service.*

Vulnerability: *Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source.*

Security Objective: *Requirements levied on an information system that are derived from laws, executive orders, directives, policies, instructions, regulations, or organizational (mission) needs to ensure the confidentiality, integrity, and availability of the information being processed, stored, or transmitted.*

Additionally, AMI-SEC utilizes the following definition in the mapping process

Security Service Domain: *A set of assets with common security concerns and requirements.*

This model captures the fact that threat agents (especially when malicious) do not always directly attack the end-target asset. The threat agent is not limited to the particular set of vulnerabilities associated with the end-target asset, but can instead exploit any vulnerability belonging to any asset within the same security service domain. The threat agent may subsequently leverage any existing and legitimate trust relationship within the domain to compromise the end-target asset. Thus, evaluation of the legitimacy or probability of a threat exploiting a specific vulnerability becomes moot.

The mapping process most importantly results in the ability to link security objectives (requirements) with security service domains. This link may subsequently be traced back through individual assets to determine appropriate mitigating controls for vulnerabilities within a specific domain.

2.3.3 Asset Identification Methodology

Assets are things of business value to the stakeholder that it desires to protect and sustain. The asset identification phase within the SRA is the first step in the assessment of critical infrastructure. Each asset identified will have a degree of due diligence applied to its risk assessment output. It is important to limit assets considered by risk management efforts to those with true value to the AMI system. Any culling of assets should occur at this early stage. To help determine asset risk, we attempt to identify its context of use, its value, its impact, and specific security concerns it may have for its use context.

2.3.3.1 Asset Identification Inputs

Inputs into the Asset Identification process can include just about anything contributing value or considered for protection. However, we are most concerned with assets having high likelihood of being compromised, high consequences resulting from compromise, or sufficient combination thereof. The list will cover assets such as:

- Business Values
- Hardware
- Software

- System Interfaces
- Data and Information
- People
- System Mission

2.3.3.2 Asset Identification Outputs

Outputs of the Asset Identification process will include:

- Description
    - Name
    - Security requirements domain
    - Asset type
    - Contexts of use
- Security Profile
    - Security concerns
    - Value
    - Impact & consequence

Asset Description
Each asset will be described by name, the security service domain in which it resides, asset type (e.g.: information, equipment, etc…), and any contextual use information that helps situate it in the AMI architecture.
Security Concerns
Protection concerns are varied as they are derived from the security attributes required by a particular system. Depending on role, location, and context an asset will have different sensitivities for each of the security attributes. These security attributes include confidentiality, integrity, availability, authentication, access control, and accounting.
Value Concerns
At the highest, most abstract level, assets are traced through business functions to organizational mission and values. The value of an individual system-level asset is ultimately derived from its role and criticality in an organization achieving said mission by the enablement of associated business functions.
Impact & Consequence Concerns
Consequence is the result of an unwanted incident, caused either deliberately or accidentally, which affects the assets. The consequences could be the destruction of certain assets, damage to the IT system, and loss of confidentiality, integrity, availability, accountability, authenticity or reliability. Possible indirect consequences include financial losses, and the loss of market share or company image.
Impact is a measurement of the magnitude of influence associated with results of an unwanted incident. The measurement of impacts permits a balance to be found between the results of an unwanted incident and the cost of the safeguards to protect against the unwanted incident. [SSE-CMM v.3]

The following table highlights a suggested classification of consequence severity due to expected asset impact based on an ANZ 4360:2004 example:

Table 1 – Example policy for consequence severity determination

		Consequence Types			
		Project Cost	**Financial Impact**	**Customer Impact**	**Regulatory and Compliance Impact**
Severity Level	High	$3M or more	$50M or more	10,000 or more	Substantial financial penalties
	Medium	$1M - $3M	$1M-$49M	1,000 to 9,999	Limited financial penalties
	Low	$1M or less	$1M or less	Less than 1,000	No regulatory or compliance issues

These consequences are provided as an example. Each utility will need to define its own thresholds for severity and impact.

Mission criticality is defined as the extent to which a system is an integral, functioning part of the business and mission of the organization. NIST has identified three categories of criticality that can be assigned to specific systems. Criticality can be interpreted as the impact on the system operation, on human lives, on operational cost and other critical factors, when a leveraged function is compromised, modified, or unavailable in the operational environment.

Table 2 – Criticality Categories

Category	Definition	Criteria
Mission Critical	Systems that would preclude an organization from accomplishing its core business functions if they fail.	Supports a core business function. Single-source of mission-critical data. May cause immediate business failure upon system failure
Important	Systems that would preclude an organization in the short term from accomplishing its core business functions if they fail.	Backup source for critical data. Extended period of time.
Supportive	Effectiveness and efficiency issues. Failures affect day-to-day business operations.	Cause loss of business efficiency and effectiveness. Tracks/calculates data for convenience.

2.3.4 Threat Assessment

A threat can be defined as a potential violation of a security mechanism. It is possible to classify threats into four broad classes [SHIREY00]:

- **Disclosure** – Unauthorized access to information
- **Deception** – Acceptance of false data
- **Disruption** – Interruption or prevention of correct information
- **Usurpation** – Unauthorized control of some part of the system

The following security services counter these threats [BISHOP02]:

- **Authentication** – Ensures that device, system, or user access is strongly mutually authenticated.
- **Authorization** – Ensures that access levels are authorized based upon strong mutual authentication. (This function is addressed within the AMI-SEC security service of Access Control.)
- **Confidentiality** - Ensures that data is shared only with authorized individuals on a need-to-know basis, and that intentional or unintentional disclosure of the data does not occur.
- **Integrity** - Ensures that data is authentic, correct and complete, and provides assurance that the data can be trusted.
- **Availability** - Ensures that data, applications and systems are available to those who need them when they need them.

Sometimes, non-repudiation is also included as a component of information security [PARKER02]. Non-repudiation refers to the assurance that a person who claims or is claimed to have created, modified, or transmitted data is in fact that person, and is unable to deny that they are responsible for the data's content or transmission.
In essence, non-repudiation is about tying a specific actor to a specific action in an undeniable manner. This function is accommodated by the AMI-SEC security service of Accounting.

2.3.4.1 Threat Model Development

A threat model is a description of a set of possible attacks to consider when designing a system. Furthermore, the threat model can be used to assess the probability, severity, and reasoning of certain attacks and allow for designers to implement proper controls for mitigation purposes. The development of a threat model includes listing the security assumptions, threat agents, motivations, threats, vulnerabilities, controls, and assets in the system of interest. Figure 2 - A Generic Threat Model shows the interaction of some of these functions.

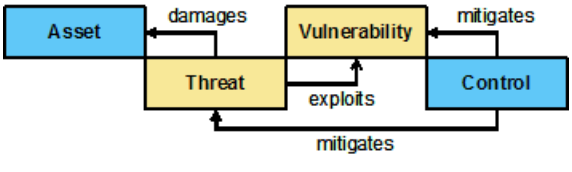

Figure 2 - A Generic Threat Model

2.3.4.2 Threats and Threat Agents

Threat agents are characterizations of entities that may have the motivation, opportunities, or means for compromising an advanced metering system. Threat agents are used to represent individuals or groups that can manifest a threat [OWASP]. These agents may be classified using four criteria:

- Objectives – The end-goal(s) of the threat agent.
- Access – The ability of the attacker to gain physical or logical proximity to the system, as well as any inherent trust assumptions.
- Resources – The financial, temporal, or manpower assets available to the threat agent.

- Expertise – The threat agent's understanding or expertise in the advanced metering infrastructure system, the electric power system, and/or the network technologies deployed by such systems.
- Risk Aversion Profile – The threat agent's tolerance for consequences that differ from the general population (e.g.: arrest, publicity, safety, etc…).

The following table gives examples of some possible threat agents [OWASP]:

Threat Agents	
Non-Target Specific	Non-Target specific Threat Agents are Computer Viruses, Worms, Trojan Horses and Logic Bombs.
Employees	Staff, Contractors, Operational and Maintenance Staff, Security Guard who are annoyed with the company.
Organized Crime and Criminals	Criminals target information that is of value to them, such as bank accounts, credit cards or intellectual property that can be converted into money. Criminals will often make use of insiders to help them.
Corporations	Companies engaged in offensive Information Warfare. Partners and Competitors come under this category.
Human Unintentional	Accidents, Carelessness
Human Intentional	Insider, Outsider
Natural	Flood, Fire, Lightning, Meteor, Earthquakes

Additionally, other non-deliberate threat agents are possible, including natural disasters, environmental and mechanical failure, as well as inadvertent actions of an authorized user may be considered [NIST80082]. This study will not consider these from an information systems security viewpoint, but should be examined in the disaster recovery and business continuity planning.

Threats are the means through which the ability or intent of a threat agent to adversely affect the advanced metering infrastructure system can be carried out [SHIREY00]. Threats are different from threat agents in that they do not necessarily imply intent. Possible threats include:

- Brute Force - Performing an exhaustive search of all possible values for a security credential or attribute (e.g. key, password or passphrase)
- Bypass - Bypassing system security functions and mechanisms.
- Destruction - Causing the destruction of system data, business data or configuration information.
- Disclosure - Losing data confidentiality.
- Denial of Service - Overloading the network and/or system resources.
- Hijack - Commandeering one-side of an existing authenticated connection.
- Malware - Deploying malicious software developed for the purposes of doing harm to a computer system or network (e.g. viruses, Trojan horses, backdoors, etc).
- Man In the Middle - Inserting undetected between two connections, where the attacker can read, insert and modify messages at will.
- Physical - Causing physical damage to or destruction of an asset.

- Privilege Escalation - Causing an unauthorized elevation of privilege.
- Replay – Creating an unauthorized replay of captured traffic.
- Repudiate - Refuting an action or association with an action.
- Sniff - Performing unauthorized traffic analysis.
- Social Engineering - Manipulating knowledgeable entities to gain privileged information or access.
- Spoof - Impersonating an authorized user or asset.
- Tamper - Modifying, in an unauthorized manner, system data, business data or configuration information.

This document will use three steps to analyzing threats:
Step 1 - determine threat-sources.
Step 2 - determine if threat sources have motivation, resources, and capabilities to carry out a successful attack.
Step 3 - apply a qualitative value to a successful attack (results of Step 2) taking into account likelihood of occurrence and impact per occurrence.

2.3.4.3 Threat Agent: Motive

Motivation can be defined as an attacker's purpose or intent to cause a desired effect on the advanced metering system. There are a variety of attacker 'attitudes' that impact individual motives, and thus vary the risk to the advanced metering system. The lack of motive reduces the likelihood that an attack will be executed. Possible motivations include:

1. Profit
    - **a.** Avoid Billing
    - **b.** Derive Revenue
    - **c.** Directly Profit
        - **i.** Resell AMI Hosted BotNet
    - **d.** Manipulate the Energy Market
    - **e.** Manipulate Unrelated Market
    - **f.** Manipulate the Economy
2. Revenge
    - **a.** Defame Individual
    - **b.** Degrade Revenue
    - **c.** Degrade Corporate Image
    - **d.** Degrade Service Delivery
    - **e.** Degrade Infrastructure
    - **f.** Extortion
    - **g.** Degrade Billing Integrity
3. Privacy / Secrecy

- **a.** Maintain Confidentiality
- **b.** Become Anonymous
- **c.** Mask Behavior
- **d.** Spoof Behavior
- **e.** Become Unobservable
- **f.** Deter Meter Deploy
- **g.** Delay Meter Deploy

4. War
- **a.** Degrade Infrastructure
- **b.** Degrade Dependent Infrastructure
- **c.** Degrade Service Delivery
- **d.** Degrade Economy

5. Ego
- **a.** Achieve Bragging Rights
- **b.** Prove Something
- **c.** Publish

6. Spying
- **a.** Degrade Confidentiality
- **b.** Reconnaissance
  - **i.** Capability Assessment
  - **ii.** Economic
  - **iii.** Technological
- **c.** Determine Operational Advantage
- **d.** Determine Market Advantage

7. Curiosity
- **a.** Explore
- **b.** Understand

8. Civil Disobedience
- **a.** Degrade Infrastructure
- **b.** Vandalism

9. Activism
- **a.** Exploit
  - **i.** Manipulate Attention to Specific Issue
  - **ii.** Manipulate Attention to Broad Issue

iii. Manipulate Attention to Unrelated Issue

b. Degrade Service Delivery

c. Vandalism

Consider impact alignment with motive
Asset integrity impact
Asset availability impact
Asset confidentiality impact

2.3.4.4 Threat Agent: Means (Capability)

Attack cost
Complexity of the Attack
Exploit availability
Time Factors of Attack
Special skills required to carry out the attack

2.3.4.5 Threat Agent: Opportunity

Access requirements
    Physical Proximity Required
    Trust requirements
Circumstantial requirements
Current Treatment of Vulnerability

2.3.5 Vulnerability

Vulnerabilities are weaknesses in the AMI system assets which increase asset exposure to attacks. Vulnerabilities stem from requirements, design, or implementation defects in the AMI system. Many general application vulnerabilities are available at the [OWASP] site.

- 3rd Party Network - Unauthorized access to the advanced metering system via a 3rd party network.
- Abuse – misuse by a valid user
- API Abuse - The most common forms of API abuse are caused by the returner failing to honor its end of this contract, returning erroneous data.
- Authentication - Weakness in the authentication mechanisms.
- Coarse Access Control - Access controls that do not allow for proper separation of duties or desired granularity.
- Code Permission - Software that requires unnecessarily elevated privileges for normal operation.
- Code Quality - Poor code quality that leads to unpredictable behavior, poor usability, and low assurance.
- Cryptographic Vulnerability – insecure, incorrect, or improperly implemented algorithms
- Dangerous API - Use of an Application Programming Interface that has known vulnerabilities, is no longer supported, or does not meet system requirements.
- Enforcement – lack of policy enforcement / assurance

- Error Handling - Improper error handling that can or does cause unintended or unpredictable behavior.
- Fail-Open: Systems should fail only into secured states (fail-secure), and never fail-open.
- Input Validation - Input that is not validated for proper formatting and content.
- Logging and Auditing - Poor or inadequate recording, retention, and handling of events of interest.
- Misconfiguration – gap between having security features and using them properly / effectively
- Protocol - Use of unknown/unproven protocols or protocols with known weaknesses inappropriate for system design.
- Sensitive - Inadequate protection of data value in transit, storage, and processing.
- Seperation of Privileges – Failure to use privilege seperation
- Services - Unnecessary services enabled on system components.
- Synchronization and Timing – improper design leads to weakness in synchronization and timing subsystems. E.g. clock manipulation,

Session Management - Inadequate session identifiers, often leading to replay attacks.

- Likelihood

2.3.6 Risk Determination

System stakeholders are highly concerned with denying or handling consequence of specific attacks on system assets. To understand the risk associated with a given concern, various factors may be taken into consideration including monetary value. The likelihood and consequence of attack to the asset stakeholder should be the primary concerns to the system builder. At high levels, these factors are easily and effectively described through subjective ranking factors and are easily derived from asset protection and classification requirements.

We provide a first rough qualitative assessment of risk due to attack or perceived vulnerability by assessing summary attack likelihood and attack consequences. Additional considerations or tables may be made to derive summary likelihood or consequence; however, in the risk assessment, the *summary* rating of a threat event against a specific asset is used.

Likelihood is summarized on a subjective scale from A to E with A being the most certain and E being rare. Consequence is summarized on a subjective scale from 1 to 5 with 1 being negligible consequence and 5 being severe consequence. Certain combinations of likelihood and consequence result in a subjective risk rating selected from low (L), medium (M), High (H), and extreme (E). A policy is first deployed for interpreting the component subjective values and subsequent assignment of risk ratings to various likelihood/consequence combinations. See **Error! Reference source not found.** for an example subjective rating interpretation policy. See **Error! Reference source not found.** for an example risk assignment policy. It is expected that specific risk ratings generate minimal due-diligence requirements for management of controls against the threat and threat sources.

2.3.6.1 AMI-SEC Likelihood Interpretation Policy

Likelihood is determined qualitatively by determining the threat agent's means, motive, and opportunism. This matrix below shows an example of a possible means for determining a

likelihood interpretation policy. Note that if any one component of *motive*, *means* or *opportunity* does not exist then likelihood is negligible.

Motive	Means	Opportunity	Likelihood
Low	Low	Low	Rare
Low	Low	High	Possible
Low	High	Low	Possible
Low	High	High	Likely
High	Low	Low	Possible
High	Low	High	Likely
High	High	Low	Likely
High	High	High	Almost Certain

2.3.6.2 AMI-SEC Consequence Interpretation Policy

2.3.6.3 AMI-SEC Risk Interpretation Policy

Table 3 – Example: Qualitative Risk Assessment Interpretation

Consequence	
1	Negligible - no impact/consequence
2	Minor - would threaten an element of the function
3	Moderate - necessitating significant adjustment to overall function
4	Major - would threaten functional goals / objectives
5	Sever - would stop achievement of functional goals / objectives

Likelihood	
A	Almost Certain - expected in most circumstances
B	Likely - will probably occur in most circumstances
C	Possible - could occur at some time
D	Unlikely - not expected to occur
E	Rare - exceptional circumstances only

Table 4 - Example Risk Rating Policy

		Consequences				
		Negligable 1	Minor 2	Moderate 3	Major 4	Severe 5
Likelihood	A (Almost certain)	M	H	H	E	E
	B (Likely)	M	M	H	H	E
	C (Possible)	L	M	M	H	E
	D (Unlikely)	L	M	M	M	H
	E (Rare)	L	L	M	M	H

E	Extreme Risk: Immediate action required to mitigate the risk or decide not to proceed
H	High Risk: Action should be taken to compensate for the risk
M	Moderate Risk: Action should be taken to monitor the risk
L	Low Risk: Routine acceptance of the Risk

2.4 Risk Assessment (Fold into the above section)

2.4.1 Introduction

2.4.2 Vulnerabilities

2.4.3 Assets

Assets are the items of protection, the target of threats, the possessors of exposures, and the beneficiaries of controls [JAQUITH07]. System assets can be defined as any software, hardware, data, administrative, physical, communications, or personnel resource within an information system [CNSS4009]. Similarly, it is possible to define assets as information, resources, or services:

1. Information Assets
    - **a.** Audit Data
    - **b.** Information Object
    - **c.** Policy
    - **d.** Other Configuration Information
    - **e.** Locally Protected Information
    - **f.** Traffic Flow
2. Resource Assets
    - **a.** AMI Virtual Network
    - **b.** AMI components
        - **i.** Software
        - **ii.** AMI applications
        - **iii.** Operating System
        - **iv.** Hardware

   c. Tokens
3. Service Assets
   a. Order Key Service
   b. Deliver Key Service
   c. Track and Control Keys Service
   d. Membership Management Service
   e. Initialization Service
   f. Software Download Service
   g. Configured Cryptographic Element Interface Service
   h. Policy Imposition Service
   i. Trust Anchor Service
   j. Network Infrastructure Services
   k. Primary Security Services
      i. Access Control Services
      ii. Integrity Services
      iii. Confidentiality Services
      iv. Accountability Services
      v. Identification, Authentication, and Authorization Services
      vi. Availability Services
      vii. Audit Services
   l. System Enrollment Services

It is important to note that each of the above assets include user data and the protection mechanisms.

2.4.4 Attacks

An attack is an attempt to gain unauthorized access to an information system's services, resources, or information, or the attempt to compromise an information system's integrity, availability or confidentiality. An attack implies intent due to the definition as an attempt. However, not all attempts are malicious.

Attacks upon the security functions themselves are called direct attacks. All assets are subject to this type of attack. Most malicious direct attacks (other than denial of service attacks) target authentication and access control mechanisms first, since defeating those mechanisms may yield additional system privileges and may provide a platform from which to launch additional attacks.

Attacks upon external entities that occur over advanced metering interfaces are called forwarded attacks. For example, an external entity floods the advanced metering network with more traffic than was allocated to the particular component—this may result in a denial of service on the network.

A third type of attack is a system attack. This sort of attack happens when the system itself, without prompting from an external user, attacks internal or external assets. This would usually occur only in the case of a malicious developer or serious hardware/software failure.
Adding security controls to an advanced metering system does not mean that the system will not be attacked, nor does it mean that the system will be impossible to compromise. An adversary with the necessary time, funding, and expertise can often compromise the most secure system.

2.4.5 Scenarios and Prioritization

Developing a set of attack scenarios allows for efficient application of security controls to help mitigate the defined attack vectors. The sole purpose of these controls is to reduce both the likelihood, and the impact of a successful attack. The likelihood of an attack refers the probability that this attack vector would be used. The impact of an attack refers the financial, reputation, or other business impact a successful penetration would have.
It is often beneficial to qualitatively sort possible attacks in terms of risk using both the likelihood and severity of the attack.
Each threat is given a severity, which is one of the following: Low, Medium, or High. The severity indicates the level of harm to the system if this threat were to succeed. A Low severity should result in no disclosure of information but, for example, might create an improperly or inconveniently configured system. A potential disclosure of information is an example of a Medium threat to the system security. A potential continuing disclosure of information is an example of a High threat.
Each threat is also given a likelihood, which is one of the following: Unusual, Unlikely, or Likely. In the case of a non-malicious threat, the likelihood is purely a probability of the threat occurring. In the case of malicious threats, the likelihood includes motivation to attack this way, whether the attack is coming from a user that some trust is placed in, and the gain from a successful attack. For malicious attacks, likelihood is less related to probability directly, since an attacker will attack a system at its weak point. Note that the likelihood is assigned before any protections are put in place. So, a threat of enrolling a user through unauthorized mechanisms is a Likely threat, simply because an attacker would be highly motivated to do it. In neither case does the likelihood include any mitigation factors implemented by the system or the environment. An unusual likelihood has an extremely low probability of occurrence. Unlikely threats have a low probability of occurrence. Likely threats are expected to be encountered and therefore require the strongest mitigation based on severity.
Some threats have a narrower focus than other threats. These threats were made specific because they have important implications in the system. The top threats were realized by combining threat components with assets to create threat statements. The following list of threat statements should be considered most apropos:
{Note: My concern about the threat ranking is that it is entirely subjective. Threat risk / severity should be determined via actual analysis of the threat, the cost to implement, and the result if achieved …}
The following attacks are considered HIGH risk with a HIGH severity if realized and a LIKELY degree of likelihood:

- A threat agent may attempt to shut off large population of meters.
- A threat agent may hijack or spoof one or more trusted systems.
- A threat agent may craft a denial of service attacks at the utility back-office.

The following attacks are considered MEDIUM risk with a HIGH severity if realized and an UNLIKELY degree of likelihood:

- A threat agent may try to obtain key material from the system.
- A threat agent may craft a denial of service attacks to a large population of meters.

The following attacks are considered MEDIUM risk with a MEDIUM severity if realized and a LIKELY degree of likelihood:

- A threat agent may try to obtain key material from a meter.
- A threat agent may attack the system using test development software or other field tools typically used by technicians or manufacturers.
- A threat agent may try to spoof the meter using stolen key material or as a man in the middle attack.

The following attacks are considered LOW risk:

- A threat agent may try to sniff messages in order to maliciously control or alter functionality.
- A threat agent may try to tamper with application protocols to maliciously control or alter functionality.

A threat agent may try to physically modify a meter to steal power

3 System Security Requirements

3.1 Policy Requirements

SSR.Policy.1 Access to sensitive information shall be limited to authorized users within the limits of their credentials and need-to-know.
SSR.Policy.2 Authorized administrators and users shall be held accountable for security relevant actions they perform.
SSR.Policy.3 Authorized administrators shall interpret, maintain, and oversees site security policy and develops and implements procedures assuring secure operation of the system.
SSR.Policy.4 Administrative responsibilities shall be split between multiple system administrators. The assignment of split administrative authorization is established in order to prevent unrestricted system control and to provide for “checks and balances”.
SSR.Policy.5 Administrators shall be responsible for installing, configuring, managing, and monitoring the performance of the system in accordance with its evaluated configuration and ensuring its conformance to applicable security policies.
SSR.Policy.6 Administrators shall review audit reports and take appropriate action.
SSR.Policy.7 Information domains shall not be directly connected without application of appropriate boundary enforcement and filtering techniques.
SSR.Policy.8 Administrators shall issue security relevant security hardware and software, and will maintain all records regarding distribution of these items.
SSR.Policy.9 The level of security afforded the system shall be in accordance with what is considered prudent by the organization’s accrediting authority.
SSR.Policy.10 Users and processes must be explicitly authorized to transfer information outside the system
SSR.Policy.11 Users and processes that transfer information into the system must be explicitly authorized to do so.
SSR.Policy.12 Data collected and produced by the system shall be protected from modification.
SSR.Policy.13 The system shall be protected from unauthorized accesses and disruptions of to system functions.
SSR.Policy.14 Only authorized system administrators, security administrators, and their representatives shall administer or repair security mechanisms within the system.
SSR.Policy.15 Only personnel authorized by the accountable organization shall have access to or utilize system resources.

3.2 Functional Requirements

(Replace mechanisms with functions)

SSR.Function.1 Shall provide administrative functions such that administrative responsibilities of the system will be well defined and compartmentalized such that administrators do not automatically have access to assets, except for necessary exceptions.

SSR.Function.2 Shall provide audit log functions which prevent unauthorized access, modification, deletion or overflow conditions.

SSR.Function.3 Shall record in audit records: date and time of action, location of the action, and the entity responsible for the action. (Move below 3)

SSR.Function.4 Shall provide functions which support the establishment of a trusted path and channel within the system and between the system and a remote trusted system for the performance of security-critical operations.

SSR.Function.5 Shall provide functions which assurance that information is not disclosed to unauthorized individuals, processes, or devices.

SSR.Function.6 Shall provide secure session establishment between the system and remote systems using approved confidentiality, integrity, authentication and non-repudiation of network transmissions.

SSR.Function.7 Shall restrict user access to cryptographic IT assets in accordance with a specified user access control policy.

SSR.Function.8 Shall provide complete separation between plaintext and encrypted data and between data and keys.

SSR.Function.9 Shall provide approved confidentiality, integrity, authentication and non-repudiation functions.

SSR.Function.10 Shall provide functions which protect cryptographic data assets when they are being transmitted, either through intervening untrusted components or directly to/from human users.

SSR.Function.11 Shall provide security services and labels on import/export data that is consistent with policy (i.e. user, data source, data content, and intended audience).

SSR.Function.12 Shall provide fault tolerant functions for critical components and continue to operate in the presence of specific failures in one or more system components.

SSR.Function.13 Shall provide integrity functions for system data, user data, and hardware/software functionality.

SSR.Function.14 Shall provide functions which uniquely identity and authenticate each user of the system.

SSR.Function.15 Shall provide functions which ensure the integrity of system data, user data, and security attributes transferred or replicated within the system.

SSR.Function.16 Shall provide functions which limit system-produced unintended emanations (intelligible or not) to within a specified limit.

SSR.Function.17 Run executable code in a protected domain where the code's potential errors or malicious code will not significantly impact other system functions of other valid users of the system.

SSR.Function.18 Provide administrative tools with a capability to observe the usage of specified services or resources as necessary.

SSR.Function.19 Shall provide functions for accountability and non-repudiation of information transfer between entities.

Comment [BB1]: Need-to-know or separation of duties? The requirement builds a wall and then puts a door in it.

Comment [BB2]: Detect – but would not use this term either since it refers to assurance… would say record. Could combine with SSR.F.3

Comment [BB3]: Assurance requirement – not functional requirement.

Comment [BB4]: Maybe too specific driving toward a particular technology? Are we concerned about covering CIA & non-repudiation for all sessions? Has the risk assessment determined this to be a requirement, i.e. does this requirement cost more than the business value.
May want to define "session".

Comment [BB5]: Define "approve"; who approves? Requirement seems overly broad; Does this assume that AMI-SEC will be the approving body?

Comment [BB6]: Dictating the access control methodology? Mandetory/Descressionary

SSR.Function.20 Shall provide functions which maintain object security attributes with integrity.
SSR.Function.21 Shall provide functions which control access to resources so that lower-priority activities do not unduly interfere with or delay higher-priority activities.
SSR.Function.22 Shall provide functions which use resource quotas to limit user and service use of system resources to a level that will prevent degradation or denial of service to other critical users and services.
SSR.Function.23 Shall provide functions which allow recovery from operations by undoing an unintended operation (i.e., "rolling back") to restore a previous known state.
SSR.Function.24 Shall provide functions which provide the ability to update the software programs to patch discovered security flaws or other flaws in the program that could be exploited by an adversary.
SSR.Function.25 Shall provide functions which provide protection of a user or admin sessions to prevent an unauthorized user from using an unattended system element where a valid user has an active session.
SSR.Function.26 Shall provide functions which maintain and recover to a secure state without security compromise after power cycle, addition or removal of components, system error or other interruption of system operation.
SSR.Function.27 Shall provide functions which manage the initialization of, limits on, and allowable operations on security attributes, security-critical data, and security mechanisms.
SSR.Function.28 Shall provide functions which maintain security-relevant roles and the association of users with those roles.
SSR.Function.29 Shall provide functions which ensure that security-relevant software, hardware, and firmware are correctly functioning through features and procedures.
SSR.Function.30 Shall provide functions which provide system features that prevent, detect, and resist physical tampering of a system component, and use those features to limit security breaches.
SSR.Function.31 Shall provide functions which maintain a set of security attributes (which may include group membership, access rights, etc.) associated with individual users in addition to user identity.
SSR.Function.32 Shall provide functions which ensure the protection provided to data in the system is predicated on the secrecy of the keys not in the secrecy of the design.
SSR.Function.33 Shall provide functions which incorporate malicious code prevention procedures and mechanisms.
SSR.Function.34 Shall provide functions which maintain a set of security attributes associated with individual components in addition to component identity.
SSR.Function.35 Shall provide functions which provide policy based access control via security attributes on Users, Components, and Objects.

3.3 Environment Requirements

(Requirements which support the operations of the system by specifying the design constraints development/implementation processes and other)

SSR.Environment.1 Shall deter administrator errors by providing adequate administrator guidance.

SSR.Environment.2 Shall implement a configuration management plan. Configuration management plan assures storage integrity, identification of system connectivity (software, hardware, and firmware), and identification of system components (software, hardware, and firmware).
SSR.Environment.3 Shall fully define cryptographic components, functions, and interfaces. Ensure appropriate protection for cryptographic keys throughout their lifecycle, covering generation, distribution, storage, use, and destruction.
SSR.Environment.4 Shall manage and update system security policy data and enforcement functions, and other security-relevant configuration data, in accordance with organizational security policies.
SSR.Environment.5 Shall evaluate system methods for proper implementation including examination for accidental or deliberate flaws in code made by the developer. The accidental flaws could be lack of engineering detail or bad design, where as the deliberate flaws would include building trapdoors for later entry as an example.
SSR.Environment.6 Shall provide backup procedures to ensure that the system can be reconstructed.
SSR.Environment.7 Shall manage and update user authorization and privilege data in accordance with organizational security and personnel policies.
SSR.Environment.8 Shall provide documentation for the general user.
SSR.Environment.9 Shall manage lifecycle maintenance such that when component hardware becomes obsolete the AMI hardware/software is redesigned to support production
SSR.Environment.10 Shall provide security administration which responds to administrative issues including fixing enrollment/I&A issues.
SSR.Environment.11 Shall provide a trusted facility for initialization.
SSR.Environment.12 Shall provide an appropriate level of physical security.
SSR.Environment.13 Shall negotiate an SLA with the Backhaul network that meets the operational needs of the mission. This includes required fault-tolerant aspects of the Backhaul's system including but not limited to routers, switch, and even "back-hoe" protection.
SSR.Environment.14 Shall provide a registration/enrollment procedure that includes all trust related elements (e.g., signatures, trust chains).

Note: Write assumptions into requirements….

Appendix A: Assets Catalogue

Appendix B: Vulnerability Catalogue

Appendix C: Threats Catalogue

Appendix D: Requirements to Threats Mapping

Appendix E: Threats to Information Domain Mapping