DRAFT – Framework Core

The *Framework Core* is a term that refers to the populated content of two matrices: a Function Matrix, and a Framework Implementation Level Matrix. The tables below show the matrix shells—that is, the unpopulated (empty) matrices.

Table 1 represents the shell for the Function Matrix. The left column contains five top-level cybersecurity functions, which are based closely on functions suggested by public comments: Know, Prevent, Detect, Respond, and Recover.

Table 1: Function Matrix Shell

Function	Category	Subcategory	Informative Reference(s)
KNOW			
PREVENT			
DETECT			
RESPOND			
RECOVER			

These five functions are defined as follows:

- **Know** – Gaining the institutional understanding to identify what systems need to be protected, assess priority in light of organizational mission, and manage processes to achieve cost effective risk management goals
- **Prevent** – Categories of management, technical, and operational activities that enable the organization to decide on the appropriate outcome-based actions to ensure adequate protection against threats to business systems that support critical infrastructure components.
- **Detect** –Activities that identify (through ongoing monitoring or other means of observation) the presence of undesirable cyber risk events, and the processes to assess the potential impact of those events.
- **Respond** – Specific risk management decisions and activities enacted based upon previously implemented planning (from the Prevent function) relative to estimated impact.
- **Recover -** Categories of management, technical, and operational activities that restore services that have previously been impaired through an undesirable cybersecurity risk event.

The purpose of the Function Matrix is to hold supporting information for each of the five functions. Each function has one or more *categories*; examples of categories for the Know function are “Know the enterprise risk architecture”, “Know the enterprise assets and systems”, and “Know vulnerability”. In turn, each category has one or more *subcategories*. Examples of subcategories for the Know function’s “Know the enterprise risk architecture” category include “Understand corporate risk tolerance”, “Identify risk assessment methodologies”, and “Identify business drivers and mission”. Each subcategory has zero or more *informative references*, which are references to existing cybersecurity-related standards, guidelines, and practices that pertain to the subcategory. An example of an informative reference is ISO 31000.

During the Framework workshop in San Diego on July 10-12, there will be working sessions held to populate the Function Matrix.

Table 2 represents the shell for the Framework Implementation Level Matrix. The left column contains a the function name derived from the Function Matrix, and the next column contains a list of three roles: Senior Executive, Business Process Manager, and Operational Manager. These roles, which were driven by public comments on the Framework, represent three viewpoints into the Framework. The other columns contain what are known as FILs—Framework Implementation Levels. A *FIL* is the extent and degree to which an organization has implemented the functions, categories, and subcategories of the Framework. In essence, that represents an organization’s maturity level.

Table 2: Framework Implementation Level Matrix Shell

Function	Role	FIL 1	FIL 2	FIL 3
Function Name	Senior Executives			
	Business Process Managers			
	Operational Managers			

The purpose of the Framework Implementation Level Matrix is to reflect the cybersecurity state of an organization by viewpoint for each of the FILs. For example, part of the FIL Level 1 (FIL 1) write up for Senior Executives might say, “My organization is able to understand cybersecurity risks and can respond to those detected,.” Meanwhile, corresponding text under FIL Level 3 (FIL 3) might say, “My organization can detect, contain and respond to cyber threats so that my essential business processes are not impacted and my critical assets are not compromised.”

Based on input provided through public comments, a straw man discussion draft of the Framework Implementation Level Matrix has been created specifically to generate and focus further input in advance of and during discussions at the San Diego workshop on the Framework.