Team 10

Prevent, Mitigate, and Recover (PMR) Insight Collective Knowledge System (PICK) Software CRC Document

Version 2.0
03/02/2020

Team 10

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

Initial Release:	1.0
Current Release:	2.0
Indicator of Last Page in Document:	&
Date of Last Review:	02/26/2020
Date of Next Review:	03/02
Target Date for Next Update:	

Distribution List

This following list of people shall receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members:
- Dr. Gates
- Dr. Salamah
- Dr. Roach
- Elsa Tai Ramirez
- Peter Hanson

Customer:
- Dr. Oscar Perez
- Vincent Fonseca
- Herandy Denisse Vazquez
- Baltazar Santaella
- Florencia Larsen
- Erick De Nava

Software Team Members:
- Charlie Juarez
- Miriam Juarez
- Angelica Marquez
- Andrew Munoz
- Aaron Rodriguez

Change Summary

The following table details changes made between versions of this document

Version	Date	Modifier	Description
1.0	02/25/2020	Aaron Rodriguez	Sections 2.1. – 2.3
1.1	02/25/2020	Andrew Munoz	Sections 2.4 – 2.6
1.2	02/25/2020	Angelica Marquez	Sections 2.7 – 2.9
1.3	02/25/2020	Charlie Juarez	Sections 2.10– 2.14
1.4	02/25/2020	Miriam Juarez	Sections 2.15-2.17
1.5	02/25/2020	Miriam Juarez	Formatting, Check
1.6	03/01/2020	Aaron Rodriguez	Sections 2.1. – 2.3 Review and update

1.7	03/01/2020	Andrew Munoz	Sections 2.4 – 2.6 Review and update
1.8	03/01/2020	Angelica Marquez	Sections 2.7 – 2.9 Review and update
1.9	03/01/2020	Charlie Juarez	Sections 2.10– 2.14 Review and update
1.10	03/01/2020	Miriam Juarez	Sections 2.15-2.17 Review and update
2.0	03/02/2020	Miriam Juarez	Formatting, Final Check

Table of Contents

1. Introduction

1.1. Purpose and Intended Audience

The CRC design method focuses on the creation of highly cohesive and modular systems. The purpose of the Software CRC Document (Classes, Responsibilities, and Collaborations) is to assist the team members in identifying and determining which classes are best suited for the system, their respective responsibilities and collaborations. The classes describe real-world objects that exist in the system, and each of these classes are assigned responsibilities, and if a class must interact with other classes, then there are also assigned collaborations.

[1] E. Tai Ramirez., "Behavior Extraction and Analysis Tool", SRS, El Paso, TX, USA, August 28, 2019.

1.2. Scope of Product

The PMR Insight Collective Knowledge (PICK) System will facilitate the current node correlation process for analysts to complete their cyber security attack graphs. Currently, the analysis process takes months to complete and the system is needed to shorten the completion time to a more reasonable length. The system will save the analysts time with the automatization of log file processing along with filtering/searching functionality. The system's attack graph creator component will further save analysts time and provide a flexible structure for analysts to work off of. The purpose of PICK is to retrieve and layout log information provided by the adversarial assessment. The software will ingest the log files to the analyst's configuration and allow the analyst to produce attack graphs containing the events that transpired to provide a report for LSH.

1.3. References

[1] S. Roach et al, Software Requirements Specification, Lethality, Survivability, and HSI Directorate (LSH), 2019.

1.4. Definitions, Acronyms, and Abbreviations

The definitions in this section are given in the context of the product being developed. This intention is to assist the user in their understanding of the document.

Definitions.

Table 1: Definition of terms used in the report

TERM	DEFINITION
Actor	A representation in the use case diagram denoting external entities that interact with a system being modeled, e.g., the testbed management system.
Extend Relationship	Denotes insertion of optional behavior of another use case into the primary use case.
Generalization Relationship	Denotes a relationship between a general use case and a specific use case.
Include Relationship	Denotes the inclusion of behavior of another use case into the primary use case.
Use Case	A modeling technique that presents the basic functionality of a system and the actors that interact with each function.
Data Cleansing	Data cleansing is the removal of unwanted characters from uncleansed TMUX log file; removal of blank rows from uncleansed excel log file; and removal of blank lines from uncleansed log file.
Data Validation	Data validation is the process of inspecting data in the cleansed log files based on predefined data validation rules.

Log Entry	Splunk takes the validated log files and convert them into normalized data. The normalized data are called log entries. Users of the system can filter and edit log entries.
Significant Log Entry	A log entry selected by the user and associated with a vector. The attributes are the same as for a log entry. The system stores significant log entries. Splunk stores log entries in the normalized data files.
Timestamp	Denotes time in hours:minutes, date in month:date:year, and section in am/pm.
Text Label	Denotes a component that displays a single line of read-only, non-selectable text [2].
Text Field	Denotes a component that implements a single line of text [2].
Text Area	Denotes a component that displays multiple lines of text.
Significant log entry	Denotes a log entry that is associated to at least one vector.

Acronyms

This section lists the acronyms used in this document and their associated definitions.

Table 2: Acronyms

TERM	DEFINITION
SRS	Software Requirements Specification
UTEP	The University of Texas at El Paso
PICK	Prevent, Mitigate, and Recover (PMR) Insight Collective Knowledge System
LSH	Lethality, Survivability, and HSI Directorate
PMR	Prevent, Mitigate, and Recover
IP	Internet Address
AA	Adversarial Assessment

Abbreviations

This section provides a list of used abbreviations and their associated definitions.

Table 1: Abbreviations

e.g.	For example
i.e.	That is

1.5. Overview

The Software CRC Document is divided into two major sections: Introduction (Section 1) and Detailed Description of Component (Section 2).

Section 1 includes four subsections. Section 1.1 provides the purpose and intended audience of the document. Section 1.2 describes the scope of the product. Section 1.3 lists the references used in this document. Section 1.4 provides the definitions, acronyms and abbreviations.

Section 2 includes one subsection. Section 2.1 and subsequent subsections contain a description of the classes.

2. Detailed Description of Component

This section will list candidate classes, superclasses, subclasses, responsibilities and collaborations.

2.1. Class Log File

Description: A file of any type (i.e. csv, jpeg, pdf, etc…) that has been ingested by the system.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.1-1	Knows its own file type	R.2-1
R.1-2	Know ingestion errors (if any exist)	
R.1-3	Know all its own log content	

2.2. Class Log Entry

Description: Normalized data recovered from log files.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.2-1	Know the attributes of Log Entry Number	R.1-3
R.2-2	Know Log Entry Timestamp	
R.2-3	Know Log Entry Content	
R.2-4	Know Host, and Source	

2.3. Class Event Configuration

Description: Configuration for a project that determines the file paths and time restraints to place on log files to be ingested.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.3-1	Know the root path of log files, file path of red team log files, file path of blue team log files, file path of white team log files.	R.1-3
R.3-2	Know the time restraints on the log files.	

2.4. Class Vector

Description: Series of activities or steps that are necessary to achieve an object.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.4-1	Know vector Name.	R.2-1
R.4-2	Know vector Description	
R.4-3	Vector Configuration has Add Vector, Delete vector, Edit vector, and a vector table	

2.5. Class Graph

Description: A visual representation of a vector.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.5-1	Knows attributes of Orientation, Export Format, Interval Units, Interval, Position of Nodes, and Position of Relationships	R.4-2 R.6-1 R.6-2
R.5-2	Graph contains at least one node	

2.6. Class Node

Description: A visual representation of a significant event.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.6-1	Knows attributes of Node ID, Node Name, Node Timestamp, Node Description, Log Entry Reference (2), Log Creator, Event Type, Icon Type, Source, Node Visibility	R.9-2 R.9-1
R.6-2	A Node is part of at least one graph (5)	
R.6-3	Knows information about its relationships	

2.7. Class Filter

Description: Used to display log entries that match the filter criteria.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.7-1	Narrows down log entry search results to those that match the filter criteria	R.2-1 … R.2-4
R.7-2	Has access to log entry data to match its filter criteria	

2.8. Class Enforcement Action Report

Description: Contains error report information.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.8-1	Knows the attributes of Line Number and Error Message.	R.1-2
R.8-2	Has access to log file (1) data to acquire Line Number error information.	

2.9. Class Beat Main

Description: Establishes the association between parent and child node.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.9-1	Knows attributes of relationship ID, Parent ID, Child ID, and Label.	R.6-1
R.9-2	Has access to a parent node (6) and a child node. (6)	
R.9-3	Has a describing label of the association.	

2.10. Class Main

Description: Window in charge of handling transition to the project.

Superclass

None

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.10-1	Handle transitions to other windows	Handle transition to ProjectConfigWindow. Handle transition to NavigatorWindow.

2.11. Class ProjectConfigWindow

Description: Window in charge of handling transition to the project workspace

Superclass

Main

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations

R.11-1	Retrieve directory path of log files	Retrieve directory data from DirDialog. Send directory data to ValidationIngestionWindow. Send event date information to ValidationIngestionWindow
R.11-2	Handle event time range for the project	
R.11-3	Handle event name and description for the project	
R.11-4	Handle transition to ValidationIngestionWindow	

2.12. DirDialog

Description: Window in charge of handling directory path

Superclass

ProjectConfigWindow

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.12-1	Retrieve directory path of log files	Send directory path to ProjectConfigWindow.

2.13. ValidationIngestionWindow

Description: Window in charge of handling the validation and ingestion process

Superclass

ProjectConfigWindow

Subclasses

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.13-1	Validate log file data	Send log file data to Splunk R.1-2
R.13-2	Ingest log file data	
R.13-3	Handle log entries	
R.13-4	Handle transition to NavigatorWindow	

2.14. Splunk

Description: External software tool used in the creation of log entries.

Superclass

ValidationIngestionWindow

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.14-1	Transform log file data into log entries.	Use MongoDB as the local database R.1-3
R.14-2	Archive log entries into a local database	

2.15. MongoDB

Description: Window in charge of handling directory path

Superclass

NavigatorWindow

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.15-1	Store data from Splunk	R.14-1 Work with NavigatorWidow to exchange data
R.15-2	Handle data management being imposed by NavigatorWindow	

2.16. NavigatorWindow

Description: Window in charge of log entry handling.

Superclass

ValidationIngestionWindow

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.16-1	Sort through log entry data.	Work with GraphTableWindow to exchange data. Send data changes to MongoDB. R.4-3
R.16-2	Prepare data changes for MongoDB.	
R.16-3	Create Vectors.	

		R.7-1

2.17. GraphTableWindow

Description: Window in charge of graphing log entry in vector format

Superclass:

NavigatorWindow

Subclasses

None

Responsibilities

Responsibilities	Responsibilities description	Collaborations
R.17-1	Create vectors.	R.14-1 R.7-1 Work with NavigatorWidow to exchange data Send data changes to MongoDB.
R.17-2	Show graphical representation of vectors.	
R.17-3	Show table representation of vectors.	
R.17-4	Edit graphical representation of vectors.	
R.17-5	Edit table representation of vectors.	
R.17-6	Create relationships among nodes.	
R.17-7	Create nodes.	