ISECOM

INSTITUTE FOR SECURITY AND OPEN METHODOLOGIES

Secure Programming Guidelines Sample

Client:	
Date:	**March 30, 2014**
Class:	**INTERNAL**
Version:	**3**

Open Source Security Testing Methodology Manual

OSSTMM

www.osstmm.org

Secure Programming Guidelines

These guidelines are to serve as as a programming framework based on the method of which the security of the applications will be verified. The premise of security within this guide is the OSSTMM 4 available at www.osstmm.org.

Quick Page Guide:

Scope

All security begins with a scope. The scope describes the environment on or in which the application will operate and the interactive points of the application. The scope is the area which needs to be secured and therefore may extend past the application itself if the application requires interactions beyond itself. The scope is the foundation for determining the application's Attack Surface, which is the unprotected or uncontrolled areas of the application which allow it to be misused or maliciously controlled.

Environment

The environment is where the application will reside and run which will show how it is intended to be used. We determine the environment by how it interacts with data, people, and physical constructs. The following should be determined to classify the environment:

1. Which specific operating systems will the system be designed to run on?
2. Will it be designed for specific hardware?
3. Will it be designed to run from specific physical locations because it will require a specific type of connectivity or can it be run from anywhere?
4. Will it be designed for mobile access and connectivity?

Interactivity

Where and how the interactivity takes place will determine the areas where controls need to be focused. It may be helpful to draw a map of this interactivity.

1. Will the application accept user input through a keypad or other such input devices?
2. Will the application accept user or system input through a network device or connection?
3. Will the application be storing or reading data (even temporarily) to a permanent location like a hard drive or other permanent, removable media?
4. Will the application be storing or reading data from a distant location (Cloud, server array, website, peers, etc.).
5. Will the application be storing or reading data in RAM?
6. Will the application be interacting with peripherals (printers, sensors,etc.) over peripheral ports (USB, AUX jack, Firewire, Serial, etc.) and note if it's one-way or bi-directional?
7. Will the application be storing or interacting within the framework of another application (a web browser, client, virtual system, sandbox, etc.)?

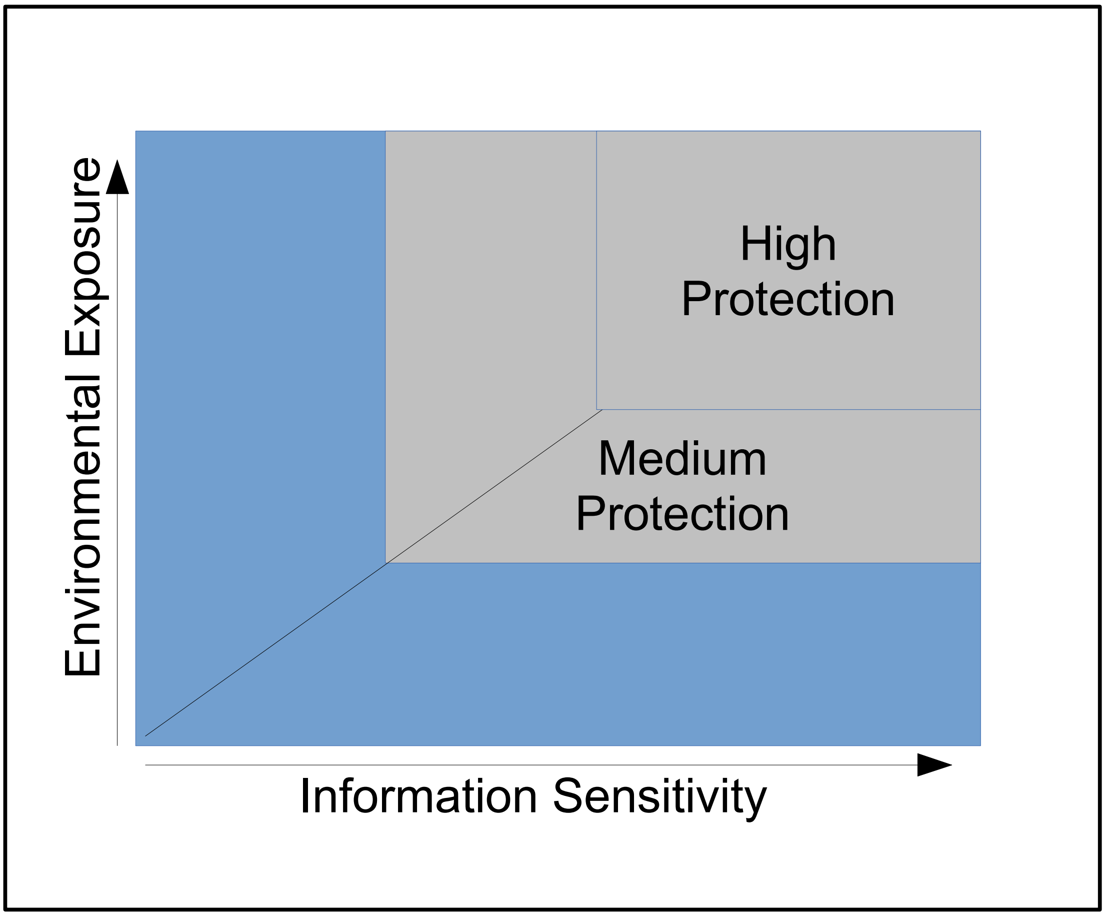
Information

The information with which the device interacts will determine how many controls are required to protect it while it is operational.

1. Will the application require storing personal or legally confidential information?
2. Will the application require storing or transacting with sensitive or confidential information in the form of logs, debugging, crash dumps, etc.?
3. Will the application be collecting or transacting with information which can personally identify a person through an aggregate of different data?

Protection Levels

An easy rule of thumb to follow is: "The more confidential the information is within the application the less exposure it should have to the environment."


Environmental Exposure
High Protection
Medium Protection
Information Sensitivity