A Design Methodology for Computer Security Testing

Marco Ramilli
http://marcoramilli.com
Arts, Crafts and Sciences

- Disciplines mature by being "arts" first, "crafts" second, and "sciences" last.

- An art is considered to be the domain of people with innate abilities and singular talents. Only someone born with a talent can be an artist.

- A craft is teachable and so requires standardized terminology, proven techniques and an established curriculum.

- To become a science, a discipline needs quantifiable measures, reproducible experiments, and established laws that make meaningful predictions.
Research Question

What approach does provide a confident measure of security in a given system?
Security Testing

National Computer Secure Center
"The portion of security testing in which the evaluators attempt to circumvent the security features of a system."

Wikipedia
"Is a method of evaluating the security of a computer system or network by simulating an attack from a malicious source"

Microsoft Security Center
"A test method in which security of a computer program or network is subjected to deliberate simulated attacks"

From Dictionary (.com)
"The process of probing and identifying security vulnerabilities and the extent to which they are used to hackers advantage"

James Shewmaker
"Is a systematic probing of a system"

Encyclopedia
"A test of a network's vulnerabilities [...]"
Security Testing

The Importance
Of all the security assurance methods, including layered design, proof of correctness and software engineering environments, only the Penetration Testing is holistic in its flaw assessment. Penetration Testing finds flaws in all the layers: policy, specifications, architecture, assumptions, initial conditions, implementation software, hardware, human interfaces, configuration control, operation, product distribution, and documentation.

To keep in mind
The attacker needs only one flaw, the vendors needs to discover all the flaws
The Security Testing Historic Path

<table>
<thead>
<tr>
<th>Toolkit</th>
</tr>
</thead>
<tbody>
<tr>
<td>The implementation in a convenient package a set of testing techniques,</td>
</tr>
<tr>
<td>usually aimed at discovering specific classes of security problems.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>The security testing organization by collecting sets of best practices,</td>
</tr>
<tr>
<td>comprehensively listing items to be tested, and structuring any other</td>
</tr>
<tr>
<td>kind of useful advice.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Methodologies</th>
</tr>
</thead>
<tbody>
<tr>
<td>The most structured approach to security testing. To different extents,</td>
</tr>
<tr>
<td>every methodology defines:</td>
</tr>
<tr>
<td>- An abstract model for the system.</td>
</tr>
<tr>
<td>- An abstract model for the process of finding its vulnerabilities</td>
</tr>
<tr>
<td>- A procedure for realizing a concrete test plan from the models.</td>
</tr>
</tbody>
</table>
Security Testing Methodology

Let’s have a close look to current Methodologies
Security Testing Methodology: ISSAF
### Security Testing Methodology: OSSTMM

[Diagram showing the OSSTMM methodology with overlapping sections such as personnel, physical, information integrity, etc.]

<table>
<thead>
<tr>
<th>Channel</th>
<th>OSSTMM Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHYSSEC</td>
<td>Physical</td>
<td>Physical security testing where the channel is both physical and non-electronic in nature. Comprises the tangible element of security where interaction requires physical effort or an energy transmitter to manipulate.</td>
</tr>
<tr>
<td>SPECSEC</td>
<td>Wireless Communications</td>
<td>Comprises all electronic communications, signals, and emissions which take place over the known EM spectrum. This includes EISEC as electronic communications, SIGSEC as signals, and EMSEC which are emissions unethered by cables.</td>
</tr>
<tr>
<td>COMSEC</td>
<td>Data Networks</td>
<td>Comprises all electronic systems and data networks where interaction takes place over established cable and wired network lines.</td>
</tr>
<tr>
<td></td>
<td>Telecommunications</td>
<td>Comprises all telecommunication networks, digital or analog, where interaction takes place over established telephone or telephone-like network lines.</td>
</tr>
</tbody>
</table>
Security Testing Methodology: Black Hats
Security Testing Methodology: GNST
Which Do I Choose !?!
### Flexibility
To provide a structured means of dynamically integrating additions in the initially defined plan, leading to richer, or more specific, new plans.

### Adaptation
The concepts and models defined within a methodology should certainly be unambiguous, but this quality should not hinder the possibility to adapt them to many different variations of the real systems to be tested.

### Guidance
The methodology should offer practical guidance about what activities compose a testing session, and which tasks are needed before, during, and after each activity.
Evaluation ...

**Reporting**
The Methodology shall support the tester in the reporting activity by meaning to help him in not omitting important details, but also letting him format the information in one or more ways that are suitable for different kinds of readers (technicians, policy-makers, managers, etc.)

**Granularity**
The methodology shall captures the details only where needed, while not uselessly encumbering the testing and reporting activities, is equally important.
Evaluation ...

<table>
<thead>
<tr>
<th></th>
<th>ISSAF</th>
<th>OSSTMM</th>
<th>BHM</th>
<th>GNST</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modeling</td>
<td>+</td>
<td>=</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Planning</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Flexibility</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Adaptation</td>
<td>=</td>
<td>+</td>
<td>+</td>
<td>=</td>
</tr>
<tr>
<td>Guidance</td>
<td>=</td>
<td>=</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>Reporting</td>
<td>-</td>
<td>=</td>
<td>-</td>
<td>=</td>
</tr>
<tr>
<td>Granularity</td>
<td>+</td>
<td>=</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Key: + good coverage  
     = average coverage  
     - limited or no coverage
The Designed Methodology

Methodology's Core

- **Define Testing Goals.** Testing goals are the final targets. What kind of information we would grab for penetration testing.

- **Define Objects.** The objects we want to test.

- **Posture of the Penetrator.** Choose between: *Internal/External, Open/Closed Box, White/Gray/Black Box*

- **Flaw Hypothesis.** Which and Where vulnerabilities could be.

- **Find Evidence.** Attacks Vector (Attacks Tree).

- **Induction Hypothesis.** New Hypothesis come out from Attacks Vector.

- **Reporting.** Almost everything.
The Designed Methodology

<table>
<thead>
<tr>
<th>Posture Penetrator Tester</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Internal/External.</strong> Where is logically situated the Attacker.</td>
</tr>
<tr>
<td><strong>Open/Closed Box.</strong> Has the attacker the ability to put/write code in the tested system?</td>
</tr>
<tr>
<td><strong>White/Gray/Black Box.</strong> What the attacker knows.</td>
</tr>
</tbody>
</table>

**Examples:**

- External Closed Black Box. Ex: Foreign Attacker who doesn’t know anything about the system. (Mercenary Attacker).
- External Closed Grey Box. Ex: Everybody from Facebook PoV.
- Internal Closed Grey Box. Ex: Marketing people.
- Internal Open White Box. Ex: Software Engineers
- External Open Black Box. Ex: Outsource society President.
The Designed Methodology
The Designed Methodology: Flaw Hypotheses

**Past Experience**
”Top-to-Bottom Review”
http://www.sos.ca.gov/elections/elections_vsr.htm

**Ambiguous and Unclear Architecture**
Examples: Where are the security attributes? (like for example, permissions, user classifications and password policies) Are there any security Pattern? Are they obsolete?

**Bypass of 'omniscent' security controls.**
Example: uVote
The Designed Methodology: Flaw Hypotheses

### Incomplete Design
Example: Usually Interface-to-Interface problems (hardware-hardware, hardware-software and software-software). Sharing is the main issue.

### Deviations from Original Model
Example: Spaghetti Code
The Designed Methodology: Flaw Hypotheses

Deviations from Original Conditions
Example: "The more you get the more you want."

System Anomalies
Example:

![Server Error](image)

Operational practices.
The Designed Methodology: Flaw Hypotheses

**Development Environment**

Example: `strcpy` function or nops padding inside `exe`

**Implementation Errors**

Example:

```php
$db = new PDO('pgsql:dbname=database');
$stmt = $db->prepare("SELECT priv FROM testUsers WHERE username=:username AND password=:password");
$stmt->bindParam(':username', $user);
$stmt->bindParam(':password', $pass);
$stmt->execute();
```
The Designed Methodology: Attack Vector

How to Generate Attack Vector?
A lot of practice (hundred of sleepless nights) and not much theory to do that. Past Experience is very important and great knowledge of exploits and tools’ 'state of the art'.

Vulnerability Classes
- Flaws that give total control of the system (TC).
- Security Policy Violations (PV).
- D/Denial oF Services (D/DoS).
- Installation Dependent (IN).
- Harmless (H).
Methodology Effectiveness

- We need to prove the effectiveness of our methodology.

- Testing **Environments**:
  - Reputation Systems: Amazon, ebay, blogs.
Methodology Generality

- We need to prove the generality of our Methodology

- Testing Environments:
  - Evade Antivirus Systems: behavior based and signature based.
eVoting Systems Results:

pVote

Attack to the Governor:

INJECTED NAME, padding

---
eVoting Systems Results: pVote

- Signal Attack:

![Exploit Code](image)
eVoting Systems Results
scantegrity and remotegrity

Feedback Engine Attack:
Reputation Systems Results

Table 7.3: Frequency of repeated sentences on three real forums.

<table>
<thead>
<tr>
<th></th>
<th>xda(^1)</th>
<th>av(^2)</th>
<th>cork(^3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Messages</td>
<td>550</td>
<td>2610</td>
<td>4244</td>
</tr>
<tr>
<td>Unique meaningful sentences</td>
<td>1707</td>
<td>9667</td>
<td>4593</td>
</tr>
<tr>
<td>Se. appearing only once</td>
<td>75.2%</td>
<td>63.5%</td>
<td>67.3%</td>
</tr>
<tr>
<td>Se. repeated twice</td>
<td>22.9%</td>
<td>26.8%</td>
<td>20.2%</td>
</tr>
<tr>
<td>Se. repeated 3 times</td>
<td>1.7%</td>
<td>7.1%</td>
<td>8.6%</td>
</tr>
<tr>
<td>Se. repeated 4 times</td>
<td>0.0%</td>
<td>1.9%</td>
<td>1.8%</td>
</tr>
<tr>
<td>Se. repeated &gt;4 times</td>
<td>0.2%</td>
<td>0.7%</td>
<td>2.1%</td>
</tr>
</tbody>
</table>

Figure 7.3: Architecture of the devised comment spam filtering prototype
Anti Virus Results: Signature Based
Anti Virus Results: Signature Based

Figure 8.2: Analysis of Zeus in an executable and in a JPG file

Figure 8.4: Analysis of first part of Spreder in a JPEG file: automatic (left), manually arranged (right)
Anti Virus Results: Behavior Based

Figure 8.7: The creations of multi processes malware
Anti Virus Results: Behavior Based

Figure 8.8: Static Analysis: BullMoose Versus Multi Process Malware.

Figure 8.9: Dynamic Analysis: Original BullMoose versus the Multi-Process BullMoose.
## Conclusions

<table>
<thead>
<tr>
<th></th>
<th>ISSAF</th>
<th>OSSTMM</th>
<th>BHM</th>
<th>GNST</th>
<th>Proposed Methodology</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Modeling</strong></td>
<td>+</td>
<td>=</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td><strong>Planning</strong></td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td><strong>Flexibility</strong></td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td><strong>Adaptation</strong></td>
<td>=</td>
<td>+</td>
<td>+</td>
<td>=</td>
<td>+</td>
</tr>
<tr>
<td><strong>Guidance</strong></td>
<td>=</td>
<td>=</td>
<td>-</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td><strong>Reporting</strong></td>
<td>-</td>
<td>=</td>
<td>-</td>
<td>=</td>
<td>+</td>
</tr>
<tr>
<td><strong>Granularity</strong></td>
<td>+</td>
<td>=</td>
<td>-</td>
<td>-</td>
<td>+</td>
</tr>
</tbody>
</table>

**Key:**
- `+` good coverage
- `=` average coverage
- `-` limited or no coverage

*Table 4.1: Feature map of the security testing methodologies*
Conclusions

1. A wide penetration testing methodology review, including parameters to evaluate these methodologies.

1. A Penetration testing methodology made by keeping the best parts of the state-of-the-art methodologies.

1. An enhanced penetration testing methodology for E-Voting.

1. Some practical scenarios. Some real examples on how to apply the methodology.