Let’s Play Poker: Effort and Software Security Risk Estimation in Software Engineering

Laurie Williams
williams@csc.ncsu.edu

Another vote for…

“Everything should be made as simple as possible, but not simpler.”
--Albert Einstein
Two Kinds of Estimation

Estimation

- How many engineers?
- How long?
Estimation

Planning Poker

Protection Poker

Effort Estimation: Planning Poker

How many engineers?
How long?

Coming up with the plan

- Desired Features
- Divide by Velocity
- 5 story points/iteration
- Estimate Size
- Derive Duration
- Iteration/release plan
- June 10
- 30 story points
- Prioritize
- 6 iterations
Estimating “dog points”

- Estimate each of the dogs below in dog points, assigning each dog a minimum of 1 dog point and a maximum of 10 dog points
- A dog point represents the height of a dog at the shoulder
  - Labrador retriever
  - Terrier
  - Great Dane
  - Poodle
  - Dachshund
  - German shepherd
  - St. Bernard
  - Bulldog
What if?

- Estimate each of the dogs below in dog points, assigning each dog a minimum of 1 dog point and a maximum of 100 dog points
- A dog point represents the height of a dog at the shoulder
  - Labrador retriever
  - Terrier
  - Great Dane
  - Poodle
  - Dachshund
  - German shepherd
  - St. Bernard
  - Bulldog

Estimating story points

- Estimate stories relative to each other
  - Twice as big
  - Half as big
  - Almost but not quite as big
  - A little bit bigger
- Only values:
  - 0, 1, 2, 3, 5, 8, 13, 20, 40, 100

Near term iteration “stories”
A few iterations away “epic”
Diversity of opinion is essential!

Vote based on:
• Disaggregation
• Analogy
• Expert opinion

(Subjective) Results of Planning Poker

• Explicit result (<20%):
  – Effort Estimate
• Side effects/implicit results (80%+):
  – Greater understanding of requirement
  – Expectation setting
  – Implementation hints
  – High level design/architecture discussion
  – Ownership of estimate
Security Risk Estimation: Protection Poker

What is the security risk?

Software Security Risk Assessment via Protection Poker

<table>
<thead>
<tr>
<th>Value</th>
<th>Ease</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Low Impact</td>
<td>Lowest Priority</td>
</tr>
<tr>
<td>High Impact</td>
<td>Highest Priority</td>
</tr>
<tr>
<td></td>
<td>Difficult to Exploit</td>
</tr>
</tbody>
</table>
Computing Security Risk Exposure

<table>
<thead>
<tr>
<th>Traditional Risk Exposure</th>
<th>probability of occurrence</th>
<th>X</th>
<th>impact of loss</th>
</tr>
</thead>
</table>

Protection Poker Overview

“Diversity of ideas is healthy, and it lends a creativity and drive to the security field that we must take advantage of.”

-- Gary McGraw

- Calibrate value of “assets”
- Calibrate ease of attack for requirements
- Compute security risk (value, ease) of each requirement
- Security risk ranking and discussion
Diversity of devious, attacker thinking is essential!

Collaborative threat modeling and misuse case development.

Memory Jogger

Value Points

1. 2. 3. 5. 8. 13. 20. 40. 100

Low value

Consider the value of the "asset" when making critical decisions.

Valuable to whom?

- The Company running the software:
  - How critical is the data in the application?
  - Can the data be restored?
  - How harmful to the company?

- The Attacker:
  - What information would be useful to the attacker?
  - What can be done with the data?
  - How much can damage can the data cause?
  - What is the impact of a breach to the attacker's business?

Ease Points

1. 2. 3. 5. 8. 13. 20. 40. 100

Hard to Attack

Consider the following as some criteria for the candidates for hardest to attack:

- Story does not create any new pages or user input fields.
- Story reduces the current number of pages or user input fields.
- A user can only access the data if a signed-in user.
- Easy to prevent the attacker from obtaining data.
- Easy to prevent the attacker from gaining control.

Easy to Attack

Consider the following as some criteria for the candidates to easiest to attack:

- Story adds new pages.
- Story adds new input fields.
- Story has few (or one) role(s) with significant read, write, update authority.
- Story requires a significant change in access control (permissions).
- Story provides default user names and passwords when the product is shipped.
- Story does not enforce strong passwords.
- Story does not have any logging or logging does not identify the specific user.
### Security Risk Assessment

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Ease Points</th>
<th>Value Points</th>
<th>Security Risk</th>
<th>Ranking</th>
</tr>
</thead>
<tbody>
<tr>
<td>Req 1</td>
<td>1</td>
<td>100</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>Req 2</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Req 3</td>
<td>5</td>
<td>1</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>Req 4</td>
<td>20</td>
<td>5</td>
<td>100</td>
<td>3</td>
</tr>
<tr>
<td>Req 5</td>
<td>13</td>
<td>13</td>
<td>169</td>
<td>2</td>
</tr>
<tr>
<td>Req 6</td>
<td>1</td>
<td>40</td>
<td>40</td>
<td>5</td>
</tr>
<tr>
<td>Req 7</td>
<td>40</td>
<td>60</td>
<td>2400</td>
<td>1</td>
</tr>
</tbody>
</table>

Sum of asset value (e.g. one 20 and one 40)

### Protection Poker High Level Overview

1. Calibrate value of database tables
2. Calibrate ease of attack for requirements
3. Compute security risk of requirements
4. Security risk ranking and discussion
**Req 1: Emergency Responder**

Currently the only roles in iTrust are licensed health care professional, unlicensed health care professional (a.k.a secretarial support), administrator and patient. The need for another role has arisen: emergency responder (ER). An emergency responder is defined as follows: police, fire, emergency medical technicians (EMTs), and other medically trained emergency responders who provide care while at, or in transport from, the site of an emergency. The only capability provided to an ER is access to an emergency report for a patient which provides basic but important information such as: allergies, blood type, recent short-term diagnoses, long term, chronic illness diagnoses, prescription history, and immunization history. The patient is sent an email to notify them of the viewing of their records by an emergency responder.

**Req 2: Find qualified LHCP**

A patient has just been diagnosed with a condition and wants to find the licensed health care professionals (LHCPs) in the area who have handled that condition. The patient chooses "My Diagnoses" and is presented with a listing of all their own diagnoses, sorted by diagnosis date (more recent first). The patient can select a diagnosis and will be presented with the LHCPs in the patient’s living area (based upon the first three numbers of their zip code) who have handled this diagnosis in the last three years. The list is ranked by the quantity of patients the LHCP has treated for that diagnosis (each patient is only counted once regardless of the number of office visits).
Req 3: Update diagnosis code table

The American Medical Association has decided that beginning January 1, 2010 all diagnoses must be coded with ICD-10 rather than ICD-9CM. These new codes need to be saved for eventual use by the iTrust application.

Req 4: View access log

A patient can view a listing of the names of licensed health care professionals that viewed or edited their medical records and the date the viewing/editing occurred is displayed.
Step 1: Calibrate value of database tables

- Which iTrust database table would be least attractive to an attacker?
- Which iTrust database table would be most attractive to an attacker?
- Use your planning poker cards to assign relative point values for the “value” of each database table, giving a 1 to the least attractive.
- Circle the database tables in Table 1 and put the value points in the appropriate column.
- There are your “value” endpoints for the rest of the exercise.
- At this time, do not assign a value to all the other tables.

Step 2: Calibrate ease of attack for requirements

- Which requirement adds functionality that will make an attack easiest?
- Which requirement adds functionality that will make attack hardest?
- Use your planning poker cards to assign relative point values for the “ease” of each requirement.
  - Easy to attack: high number
  - Hard to attack: low number
- Record ease values in Table 3.
- There are your “ease” endpoints for the rest of the exercise.
- At this time, do not assign a value to all the other requirements.
Step 3: Compute security risk of requirements

- For each requirement:
  - Identify database tables used in that requirement and record in Table 2. For each:
    » Table already have a “value”? Use it.
    » Table doesn’t have a “value”? “Poker” a value and put it in Tables 1 and 2
  - Put sum of all database values in Table 3.
  - “Poker” a value for ease points for each requirement and record in Table 3.
  - Compute security risk in Table 3 by multiplying value by ease.

Step 4: Risk Ranking and Discussion

- Rank your risks.
- Any surprises? Satisfied with values you gave?
- What plans would you put in place now that you are more aware of the security risk?
Anticipated Results of Protection Poker

• Explicit result (20%):
  – Relative security risk assessment

• Side effects/implicit results (80%):
  – Greater awareness understanding of security implications of requirement
  – Allocation of time to build security into new functionality “delivered” at end of iteration (appropriate to relative risk)
  – Knowledge sharing and transfer of security information