UISA: User Interface Security Assessment
Sarah Abdellahi and Heather Lipford, University of North Carolina at Charlotte
Carrie Gates and Jeremiah Fellows, Bank of America

What is UISA?
While organizations have independent, mature processes for evaluating the usability of their software as well as security threats and risks for their assets, methods at the intersection of these two concerns are still missing.

UISA is a method for designers and developers to use to assess the security implications of a particular user interface design, based on how users will interact, and in particular, behave in unexpected and unintended ways.

Why UISA?
- Help designers evaluate the security risks of a particular design or application
- Prioritize aspects of a design that may need improvement
- Compare different design ideas based on the potential impact on security
- Facilitate structured conversations between stakeholders with different concerns related to usability and security

Future of UISA
- Evaluation of UISA process with participants with different backgrounds to improve instructions and handbook
- Case studies with different enterprise applications and designs, to evaluate handbook completeness and UISA outcomes
- Expansion with a security risk metric, to quantify the security risks of interface design decisions

UISA Guideline 5 Steps Process

**Step 1**
*Identify application workflows*, to focus the assessment on particular tasks and screens of an application.

**Step 2**
*Elaborate all of the user interface elements* that are part of those workflows. This includes elaboration of every element where users provide input, navigate or perform activities, including navigation links and tabs, menus, text entry, and checkboxes and buttons.

**Step 3**
*Identify all of the possible errors users* could make interacting with each of those individual elements, as well as possible errors they make interacting with a combination of elements in workflow level actions.

**Step 4**
*Determine the security consequences* of each error, and filter out those without security implications.

**Step 5**
*Identify the potential triggers* for the errors based on usability and usable security guidelines.

Example Task/Workflow Level Potential Error Types
- User abandons the workflow and does not finish the task
- User performs the task through non-legitimate alternatives
- User performs an unintended task instead of the original task
- User provides placeholder data or performs blind actions to move forward

Example Potential Element Level Errors for Drop Box/Combo Box
- Do not make a choice
- Choose fewer options than applicable
- Choose more options than applicable
- Choose multiple contradicting options

Example Potential Security Consequences
- Could wrong next level decisions be made due to inaccurate input?
- Could the process/workflow fail because the user performs a wrong action or cannot provide the required input?
- Could the user adopt a non-legitimate alternative solution because of the interface complexity, too much restriction or timely process?
- May data integrity be damaged due to wrong input or wrong actions leading to modification of data or wrong input?

Example Design Triggers for Element Level User Errors
- Unclear language or wrong expectation of user's knowledge and familiarity
- Design element is not visible due to visual triggers such as clutter, color, etc.
- Slips such as unintended press of the button when a keyboard key is pressed
- Multiple relevant choices exist but user is forced to choose one (Or assumes can only choose one)
- Users are forced to provide information they are not comfortable providing