Tyler Thomas

ASIDE Study Script iTrust

Draft 3

2/13/2015

Hi, My name is <Interviewer Name>. We are studying a security oriented Eclipse plugin called ASIDE, and we are currently conducting a study which examines its interface. We hope to learn what developers think of the interface, and how effective ASIDEis at aiding the developer in the detection of access control vulnerabilities. In lay terms, access control code controls the access to certain data or functionality based upon who the user is or what role they are in – such as a user only being able to view his own health information, and not other people’s. Consequently, access control vulnerabilities are vulnerabilities in which a user is allowed to access functionality that they were not intended to access or obtain data that they are not intended to obtain without the exploitation of other vulnerabilities. To get started, please sign this consent form. It basically says that participation is voluntary, your information is protected and anonymized, and we are not going to do anything unethical. Note that we are going to use audio and screen recording software.

(pass consent form and pen)

Thanks for signing that for me. Please fill out this demographic survey.

(pass survey)

Thanks for filling that out. I’m going to show you ASIDE now.

(run ASIDE, navigate to training example in webroot/auth/pha/adverseEventChart.java)

This is a training example to help you get familiar with the way ASIDE works before we observe the real application. You can see this yellow question mark over here (point to question). This is ASIDE saying that this operation looks sensitive or is on the path of an operation that looks sensitive. If you examine the operation, it looks like it is used to update the balance on an account. This code up here (highlight code) is the access control check that makes sure the user is logged in and owns the account. ASIDE is requesting that you show it where the access control is located for this sensitive operation by highlighting it. To do that, you can left click on the yellow question mark and look at the options. Please left click one time on the yellow question mark.

(wait for user to click on yellow question mark)

All of the options that don't have the yellow question marks beside them are provided by eclipse and have nothing to do with ASIDE so please ignore those. This option at the top called "Add Annotation" will allow you to highlight or "annotate" the access control code. Please double click it now

(wait for user to double left click add annotation)

Now ASIDE is ready to highlight or annotate access control checks for this sensitive operation. Please highlight the code you think is an access control check for this sensitive operation. When you are done, please click somewhere else in the code so aside will know you are finished highlighting.

(wait for user to highlight code then click somewhere else in the code so aside knows the annotation is done)

Now you can see that the yellow question mark has changed to a green check with green highlighting, and you can see that a small green diamond has appeared next to the code we just highlighted. The green check means that ASIDE is happy. It is saying that the sensitive operation is now associated with an access control check. The small green diamond simply means that this code has been marked as an access control check.

If you feel that you want to add more annotations to this check, you can do it by clicking on the green check and double clicking the "Add Annotation" option.

You can also see what code the annotation is associated with simply by clicking on the annotation, represented by a green diamond, and viewing the popup. Please click on the green diamond annotation and observe the popup.

(wait for user to click on green diamond annotation and show window)

If you double click the first option (the title), ASIDE will open the file containing the associated sensitive operation, navigate to the sensitive operation, and select for you. Like this:

(wait for user to double click the title)

Let's say that after some review. It looks like making this annotation was a mistake. Perhaps you highlighted the wrong thing. The annotation can be deleted by left clicking on the annotation, represented by a green diamond, and double clicking "delete annotation." Please do this now.

(wait for user to left click on the green diamond annotation and double click delete annotation)

Note that if ASIDE detects a problem with access control and believes it has a very good chance of being vulnerable, it will display a red flag. They are very similar to yellow question marks, and they can be annotated as such. However, ASIDE is much more confident that they are actually vulnerabilities. That's pretty much it. Please play with the traniner example, let me know if you have any questions. When you feel you understand the trainer and wish to continue, let me know and we will continue.

(answer questions. When ready, continue)

I’m going to start the recorder now.

(start recorder and screen recording software)

(navigate to auth/admin/editNDCodes.java)

This is used for editing ND Codes, which are drug names. Please use ASIDE to annotate the access control logic for the sensitive operation. If you have any questions about the code, feel free to ask.

(wait)

(Ask questions in interview printout)

(navigate to auth/admin/addER.java)

This is used for adding emergency responder users. Please use ASIDE to annotate the access control logic for the sensitive operation.

(wait)

(Ask questions in interview printout)

(navigate to auth/admin/addHCP.java)

This is used for adding health care provider users, or medical doctors. Please use ASIDE to annotate the access control logic for the sensitive operation.

(wait)

(Ask questions in interview printout)

(navigate to auth/admin/addPHA.java)

This is used for adding pharmacist users. Please use ASIDE to annotate the access control logic for the sensitive operation.

(wait)

(Ask questions in interview printout)

(navigate to auth/er/information.java)

This file is used so that emergency responders can view their own personal information. Please use ASIDE to annotate the access control logic for the last twosensitive operations.

(wait)

(Ask questions in interview printout)

Now that we are done looking at all the annotation requests. I'd like to ask a few questions about ASIDE as a whole.

(Ask last section of questions in interview printout)

I'm going to stop the recorder now.

(stop recorder)

That's it. That's all we need. Thanks so much for your time and participation in our study.

Here is your compensation

(Hand over compensation, get signature)

INTERVIEWER NOTES

1. adverseEventChart

pha folder

role check pha

role checking code present

simple example

trainer

line 26 marker

2. editNDCodes

admin folder

role check admin

role checking code present

code edits nd codes

simple example, not trainer

line 48 marker

3.addER

admin folder

role check admin

role checking code present

code adds emergency responder users

sets the stage for vulnerability detection in number 4

line 36 marker

4. AddHCP

admin folder

role check admin, not vulnerable

aside shows vulnerable

code adds health care provider workers, shows vulnerability

line 35 marker

5. addPHA

admin folder

no role checking, vulnerable

aside shows vulnerable

code adds pharmacist , shows vulnerability, role checking code not present

line 27 marker

6.er/information

er folder

Access control provided based on session at line 7 not vulnerable true positive

code allows emergency responders to view their own information

Multiple Requests

line 20 marker and line 24 marker