



AN ONLINE USER BEHAVIORAL DATA COLLECTION INFRASTRUCTURE

Johns Hopkins University Information Security Institute
June 2019

TABLE OF CONTENTS

INTRODUCTION 1
COMPONENTS OVERVIEW 2

INFRASTRUCTURE SETUPS(MANUALS).....	3
1. MAIL SERVER INSTALLING AND CONFIGURATIONS	3
Prerequisites.....	3
Step 1 — Installing Postfix	3
Step 2 — Configure MySQL.....	4
Step 3 — Configure Postfix.....	5
Step 4 — Configure Dovecot.....	5
2. MAIL CLIENT INSTALLING AND CONFIGURATIONS	6
Prerequisite:	6
Step:1 Install PHP & Apache using below command:.....	6
Step:2 Create & Configure Roundcube Database.....	6
Step:3 Download tar file of Roundcube	6
Step:4 Start Roundcube installation using web browser installer.....	6
Step:5 Access the Roundcube Login Page.....	7
Step:6 Log into Roundcube	8
3. QUALTRICS PUBLISHING AND CONFIGURATIONS	8
a) Survey Flow of Pre-survey.....	8
b) Survey Flow of MT & NO MT Survey.....	12
c) Survey Flow of Post Survey	14
DESIGN & IMPLEMENTATION (CODE SNIPPET AND COMMENTS)	17
1. Roundcube Modifications	17
a) Disable Menu Buttons	17
b) Disable Menu Buttons Dropdown	17
c) Disable Right Click.....	18
d) Disable F5.....	18
2. Adding Event Listeners, Log Format and AJAX on Roundcube	18
a) Listeners on Qualtrics Iframe.....	18
b) Listeners on Folder Buttons	19
c) AJAX Script to Send Event Capture Data.....	19
d)Click Listener on Email List	19
e) Add “Move to” to Toolbar	20
f) Add Listener to “Move to” Button	20

g) Detect Mouse Move In or Out of Roundcube UI	21
h) Capture IP Address and Geo-Location	21
i) Create New Alert UI	21
j) Enable Alert When Email List Count Reaches Zero	22
k) Browser Capability	23
3. EVENT LISTENERS, LOG FORMAT AND AJAX ON QUALTRICS	23
Official Instructional Support	23
a) Centered Timer	25
b) Get User and Question HTML Elements	25
c) Add Event Listeners to Question Body	26
d) Get All the Choice Buttons HTML Elements	26
e) Add Event Listeners to Choice Buttons	26
f) Send Logged Data to Institute Server via AJAX	27
4. INTEGRATION BETWEEN ROUND CUBE AND QUALTRICS	28
a) Presentational Level	28
b) Logical Level	28
5. CONCUREENT USER ACCESS ISSUE	29
EXPERIMENT SETUP (SCRIPT SNIPPET AND COMMENTS)	31
1. USER ACCOUNTS CREATION	31
2. EMAILS POPULATING	31
TESTING CASE REPORT	Error! Bookmark not defined.
APPENDICES	32
A. QUALTRICS SURVEY QUESTIONS	32
B. DATA DICTIONARY	32
C. INFORMED CONSENT FORM	Error! Bookmark not defined.
D EXPERIMENT USE EMAILS DESIGN	32
REFERENCES	32

INTRODUCTION

Employees in working environment are required to deal with emails as part of daily routines while phishing attacks through email communications have been more frequent and sophisticated. Research efforts have been made to propose counter measures to mitigate such risks brought by phishing emails. Specially, this project aims to study how would distractions of multitasking and material incentives would impact human judgement of phishing emails. Research related to human behaviors have been widely conducted with the presence of human subjects while our research team innovatively designs and implements a survey platform to perform human behavioral with most available open source programs and third-party gadgets. We managed to collect user behavioral data from different granularities and export such information into log files with defined format per human subject. With such data collected, we will be able to understand different user strategies applied to identify emails under different conditions and come up with measures that will help users counter such phishing threat.

This document mainly focused on the technical design and implementation of the email processing and question answering platform developed to collect user behavioral data while user is performing email sorting taskings remotely. The integrated survey platform is powered by Roundcube, a web-based mail client hosted on Institute's server, Qualtrics, a third-party browser-based survey software, Postfix, a smtp mail server, and Amazon Mechanical Turks, an online platform to post human intelligence tasks. With numerous modified features, our survey platform simulates real-world scenarios where users are required to process emails at work and enables user behavior data collecting during that process. The emails used in the experiment are based on real world enterprise emails offers and phishing emails databases and are further modified by the team. The experiment also set the participants to be under conditions of multitasking(distracted) and/or financial incentives to study the impact brought by distraction and incentive.

The data we are collecting include user operations (email classifications, clicking, hovering events of interested html elements) and timestamps of the email sorting operations together with other operational data generated by user interaction with the Qualtrics interface, except for the survey data collected by Qualtrics. The data is stored in a log file after the username of Postfix Accounts

Task Description:

(1): Distractions and Incentivization

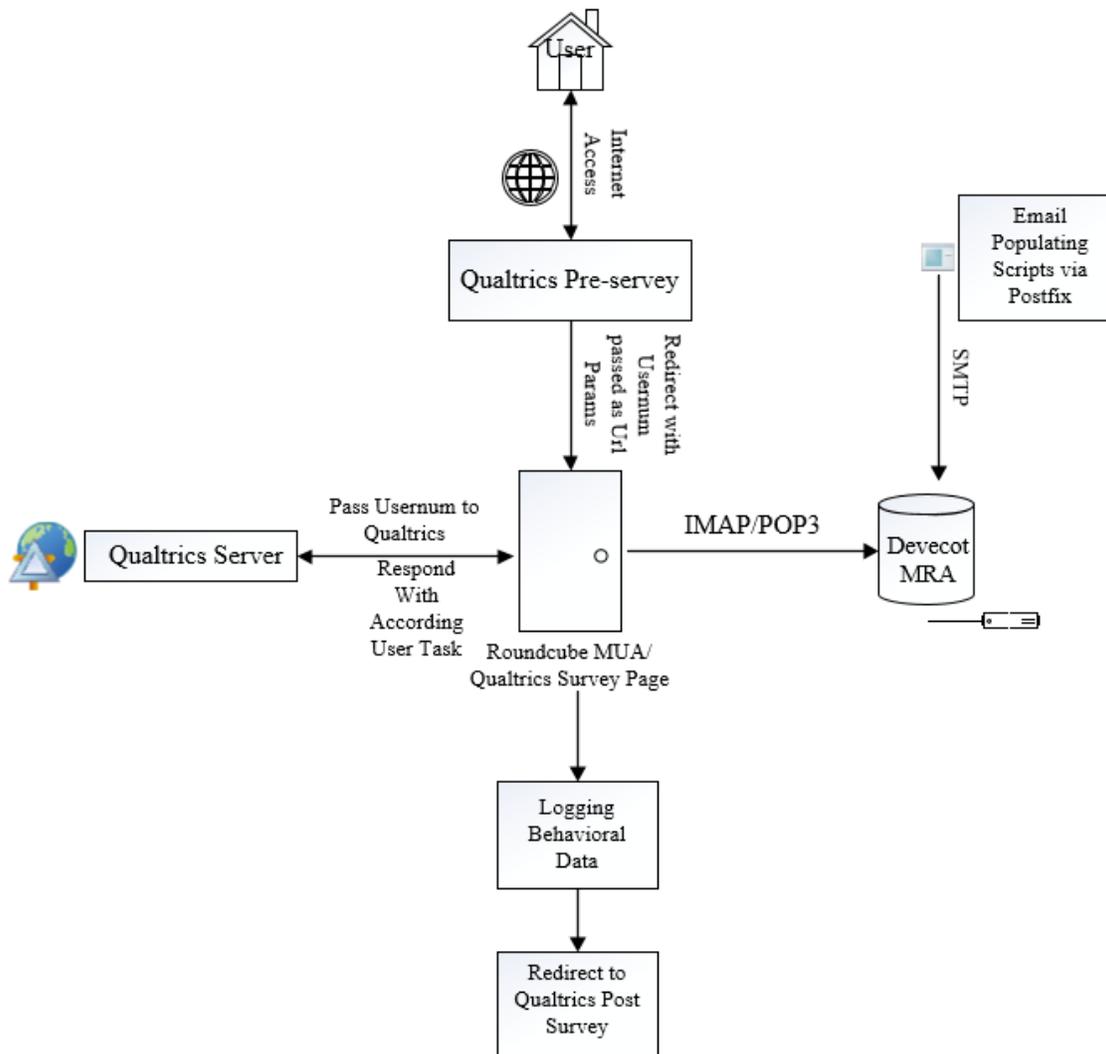
Distraction is implemented by answering questions presented on Qualtrics. Incentivization rewards those users achieving high accuracy in a tiered scheme.

(2): Conditions

According to the use of two factors in (1) one participant is randomly put in one of four conditions.

COMPONENTS OVERVIEW

The following graph describe the general workflow between the several components of the user behavior study. A dedicated email account is generated for each participant with the same naming patterns: smithj##### under the same domain name: jhu.edu. Each email account is preloaded with a batch of 40 emails via the execution of a PHP script on the SMTP server. The participant end starts from taking the pre-survey on Qualtrics and will be redirected to a mid-survey consisting of the roundcube user interface as well as Qualtrics, though their tasks may be different based on their randomly generated user condition and thus the user number. The roundcube user interface works as a MUA retrieves email items from Dovecot via IMAP/POP3 protocols. The Qualtrics user interface maintains an active session presenting the distracting tasks and/or recording the time consumed in this section. User interactive operations on both parts will be logged in real time and saved as a file on the remote server. Once the user runs out of the required time of this section or finishes early and chooses to move on, the user will be taken to the post survey on qualtrics and will not be able to come back to change their responses to the mid-survey.



INFRASTRUCTURE SETUPS(MANUALS)

1. MAIL SERVER INSTALLING AND CONFIGURATIONS

Postfix is a *mail transfer agent* (MTA), an application used to send and receive email.

Prerequisites

To install Postfix Mail Server, you will need:

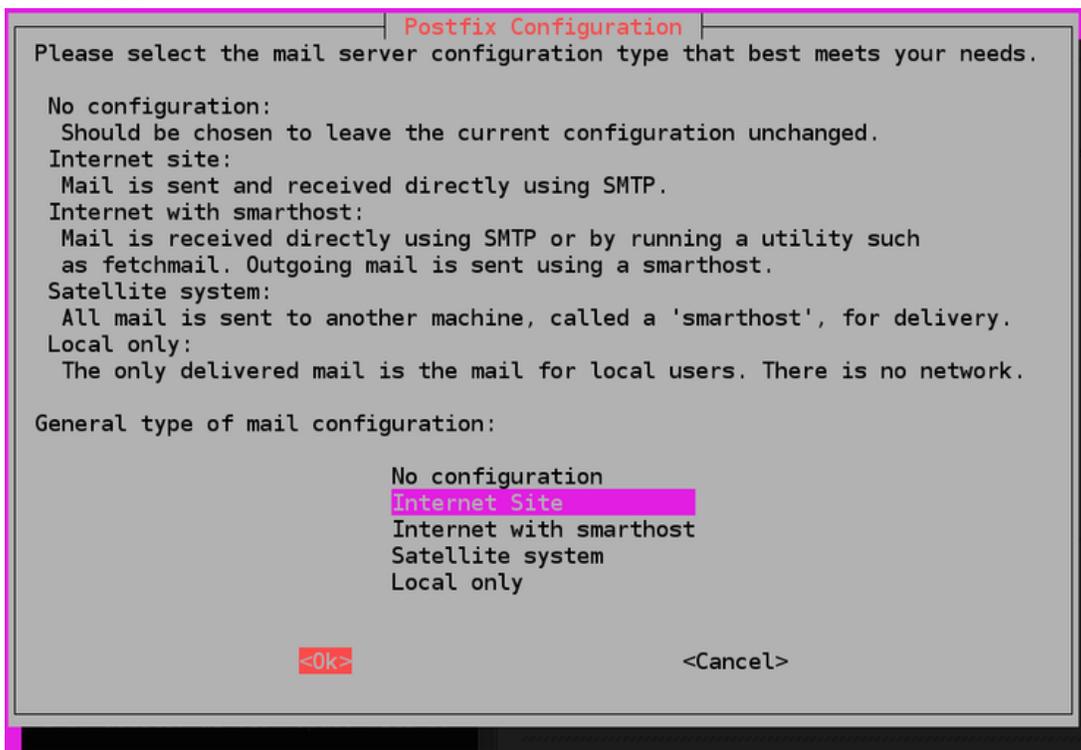
- One Linux distribution, some initial setup including creating a sudo non-root user
- A valid domain name, like example.com, pointing to your server.

Step 1 — Installing Postfix

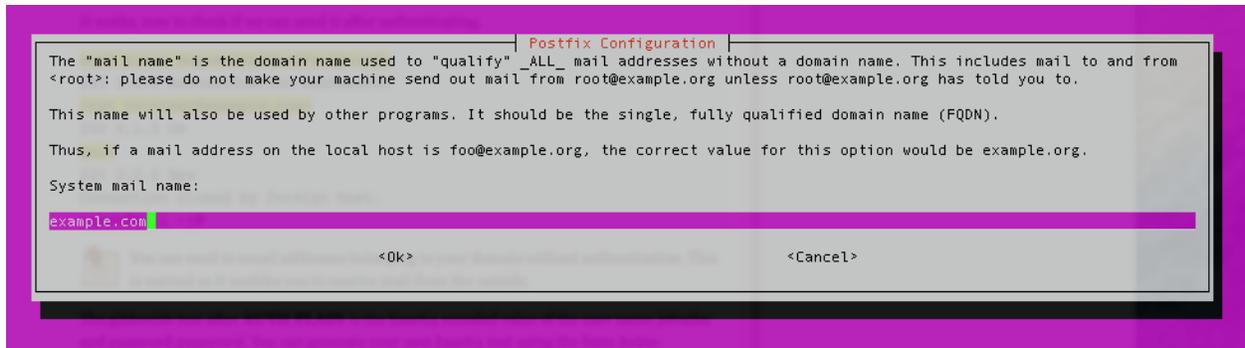
- Update the package database: `$ sudo apt-get update`
- Install Postfix as well as a few other programs needed for Postfix to function:

```
$ apt-get install postfix postfix-mysql dovecot-core dovecot-imapd dovecot-pop3d dovecot-lmtpd dovecot-mysql mysql-server
```

- Near the end of the installation, the user will be presented with the following dialog window, Internet Site is the default option that will enable emails to be sent and received via SMTP



- Set System Mail Name to be the FQDN, in our case: ahab.isi.jhu.edu



Step 2 — Configure MySQL

- Create a new database: `mysql> CREATE DATABASE Postfix;`
- Create the MySQL user and grant the new user permissions over the database. Replace password with a secure password: `GRANT SELECT ON Postfix.* TO 'mailadmin'@'127.0.0.1' IDENTIFIED BY 'password';`
- Flush the MySQL privileges to apply the change: `FLUSH PRIVILEGES;`
- Create a table for the virtual domains that will receive mail on the server:

```
CREATE TABLE `virtual_domains` (
  `id` int(11) NOT NULL auto_increment,
  `name` varchar(50) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

- Create a table for all of the email addresses and passwords

```
CREATE TABLE `virtual_users` (
  `id` int(11) NOT NULL auto_increment,
  `domain_id` int(11) NOT NULL,
  `password` varchar(106) NOT NULL,
  `email` varchar(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `email` (`email`),
  FOREIGN KEY (domain_id) REFERENCES virtual_domains(id) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

- Create a table for the email aliases:

```
CREATE TABLE `virtual_aliases` (
  `id` int(11) NOT NULL auto_increment,
  `domain_id` int(11) NOT NULL,
  `source` varchar(100) NOT NULL,
  `destination` varchar(100) NOT NULL,
  PRIMARY KEY (`id`),
  FOREIGN KEY (domain_id) REFERENCES virtual_domains(id) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

- Add the domains to the virtual_domains table. Replace the values for example.com and hostname with your own settings.

```
INSERT INTO `Postfix`.`virtual_domains`
(`id`, `name`)
VALUES
('1', 'ahab.isi.jhu.edu),
('2', 'jhu.edu),
```

Step 3 — Configure Postfix

- Make a copy of the default Postfix configuration file in case you need to revert to the default configuration: `cp /etc/postfix/main.cf /etc/postfix/main.cf.orig`
- Edit the `/etc/postfix/main.cf` file to match the [main.cf](#)
- Create the `/etc/postfix/mysql-virtual_domains.cf`. Ensure that you change the password for the Postfix user account, which should match the [mysql-virtual_domains.cf](#)
- Create the `/etc/postfix/mysql-virtual-email2email.cf` file. Ensure the file matches with [mysql-virtual_email2email.cf](#)
- Create the `/etc/postfix/mysql-virtual-mailboxes.cf` file. Ensure the file matches with [mysql-virtual-mailboxes.cf](#)
- Create the `/etc/postfix/mysql-virtual_forwardings.cf` file. Ensure the file matches with [mysql-virtual_forwardings.cf](#)
- Edit the `/etc/postfix/master.cf` file to match the [master.cf](#)
- Change the permissions on the `/etc/postfix` directory to restrict permissions to allow only its owner and the corresponding group: `chmod -R o-rwx /etc/postfix`
- Restart Postfix: `service postfix restart`

Step 4 — Configure Dovecot

- Open the main configuration file: `/etc/dovecot/dovecot.conf` to match [dovecot.conf](#)
- Open the `/etc/dovecot/conf.d/10-mail.conf` file to match [10-mail.conf](#). This file controls how Dovecot interacts with the server's file system to store and retrieve messages.
- Create the `/var/mail/vhosts/` directory and a subdirectory for the domain names:

```
[root@ahab vm]# ls
ahab.isi.jhu.edu dovecot-deliver.log jhu.edu
```
- Create the `vm` user with a user and group id of 5000: `groupadd -g 5000 vm; useradd -g vm -u 5000 vm -d /var/mail`

```
[root@ahab vm]# cat /etc/group | grep vm
vm:x:5000:
```
- Change the owner of the `/var/mail/` folder and its contents to belong to `vm`: `chown -R vm:vm /var/mail`
- Edit the `/etc/dovecot/conf.d/auth-sql.conf.ext` file to match [auth-sql.conf.ext](#)
- Change the owner and group of the `/etc/dovecot/` directory to `vm` and `dovecot`: `chown -R vm:dovecot /etc/dovecot`
- Change the permissions on the `/etc/dovecot/` directory: `chmod -R o-rwx /etc/dovecot`
- Open the sockets configuration file, located at `/etc/dovecot/conf.d/10-master.conf` to match [10-master.conf](#)
- Open `/etc/dovecot/conf.d/10-ssl.conf` to enable `ssl` which should match [10-ssl.conf](#)
- Finally, restart Dovecot: `$ service dovecot restart`

2. MAIL CLIENT INSTALLING AND CONFIGURATIONS

Prerequisite:

- PHP
- Database (MariaDB or MySQL)
- Web Server (Apache)
- Any MTA – Postfix

Step:1 Install PHP & Apache using below command:

```
yum install httpd php php-common php-json php-xml php-mbstring php-imap php-pear-DB php-mysql
```

Once the above package list is installed, set the time zone value in PHP.

```
$vim /etc/php.ini
```

```
date.timezone = America/New_York
```

Save & exit the file

Step:2 Create & Configure Roundcube Database

- Create a new database: `mysql> CREATE DATABASE webmail;`
- Create the webmail user and grant the new user permissions over the database. Replace password with a secure password: `GRANT SELECT ON webmail.* TO 'roundcube'@'127.0.0.1' IDENTIFIED BY 'password';`
- Flush the MySQL privileges to apply the change: `FLUSH PRIVILEGES;`

Step:3 Download tar file of Roundcube

- Download latest version of Roundcube tar file either from their official site <https://roundcube.net/download/>
- Rename the extracted file as Roundcube and set the required permissions:

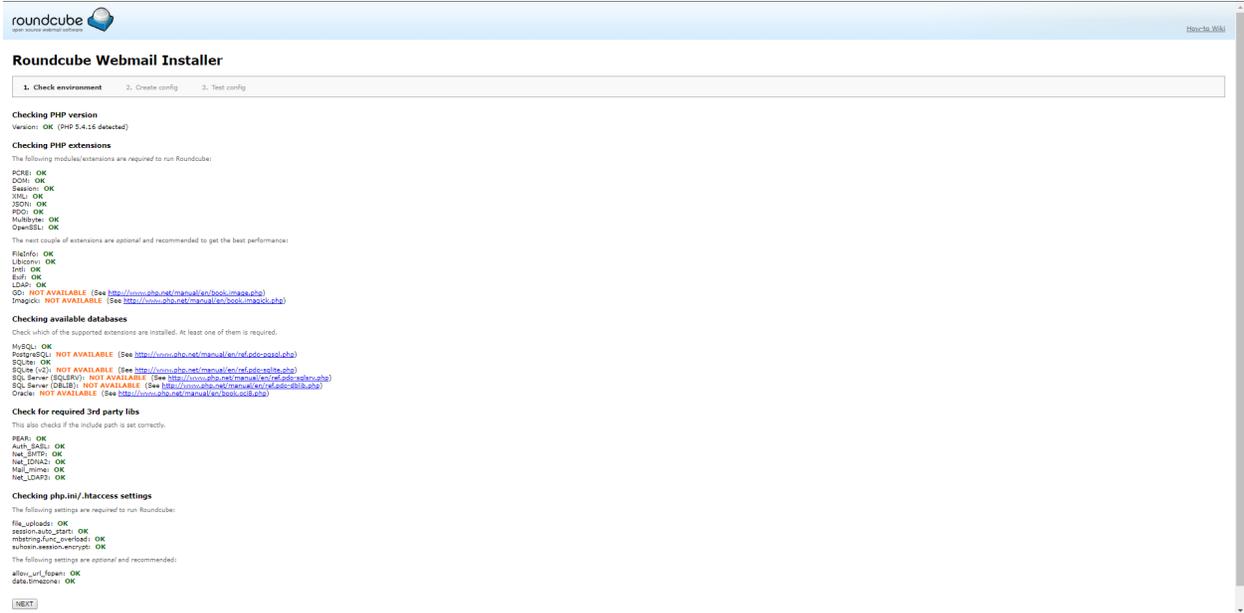
```
$ mv roundcubemail-1.2.7 roundcube
```

```
$ chown apache:apache roundcube
```

Step:4 Start Roundcube installation using web browser installer.

- To start the installation of Roundcube, access the below url in the web browser:

```
http://ahab.isi.jhu.edu/roundcube/installer
```

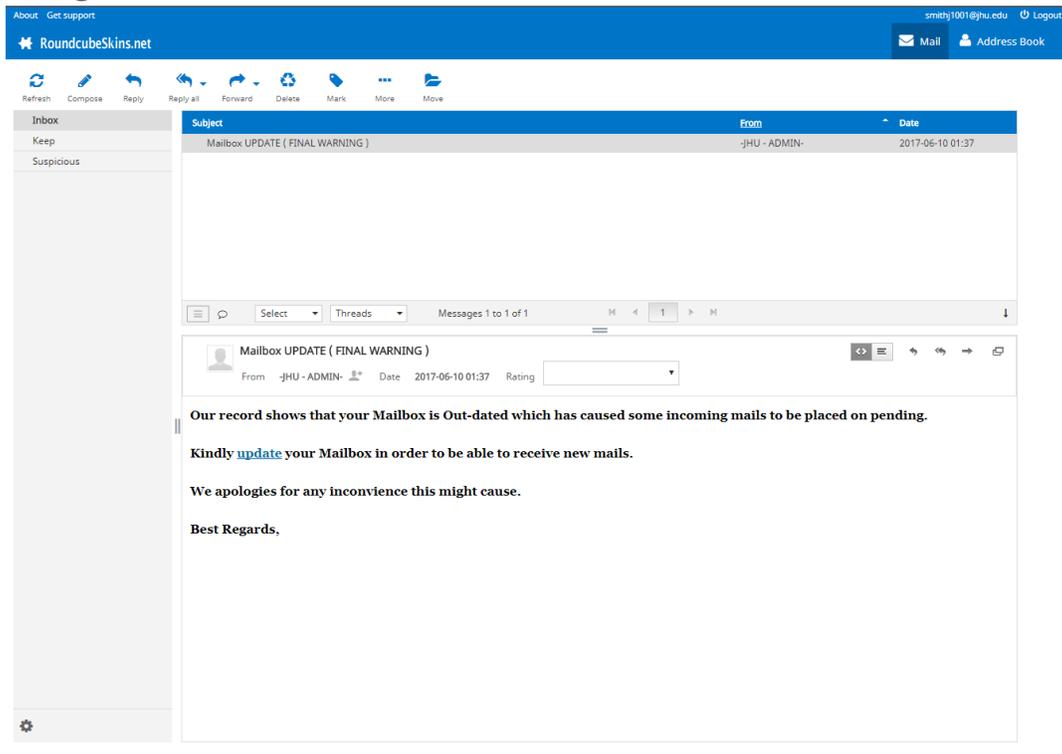


- Define the Database, Authentication Mechanism, SMTP & IMAP setting in this step.
- Make sure the config.inc.php file within the /var/www/html/roundcube/config/ match with [config.inc.php](#)
- Remove the installer directory from your apache server document root

Step:5 Access the Roundcube Login Page



Step:6 Log into Roundcube



3. QUALTRICS PUBLISHING AND CONFIGURATIONS

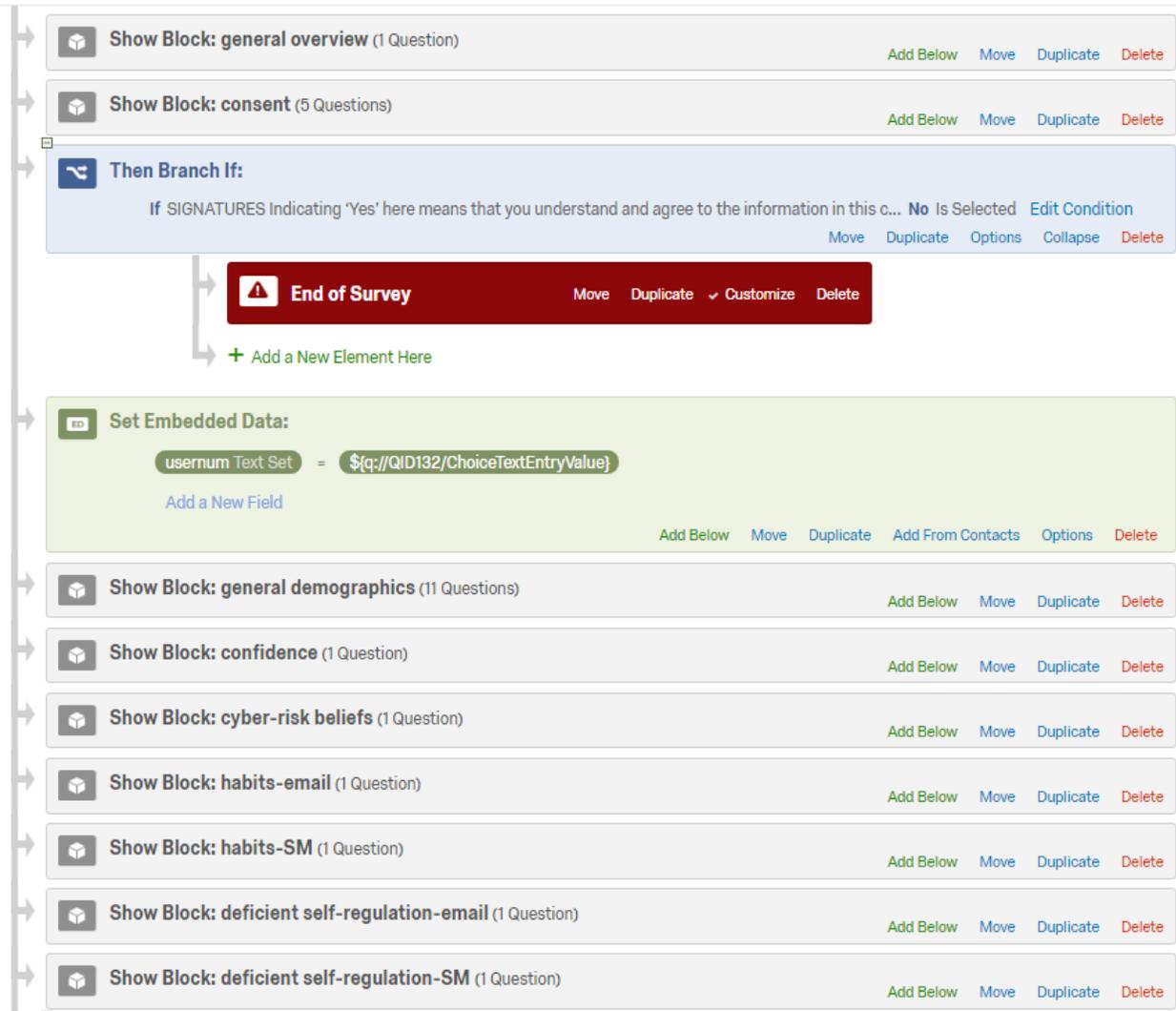
a) Survey Flow of Pre-survey

The pre-survey subscribes to a web service provided by qualtrics that returns a random number from 1 to 4 and it is stored as embedded data in pre-survey as “condition” field. “MID” is another embedded data field that stores the Mechanical Turks ID passed through URL parameters. The pre-survey then subscribed to a web service provide by the institute server that returns a user number after providing the random generated “condition” value. The returned value is identified as the participant ID and stored as “pid” and “usernum1” fields in the embedded data type.

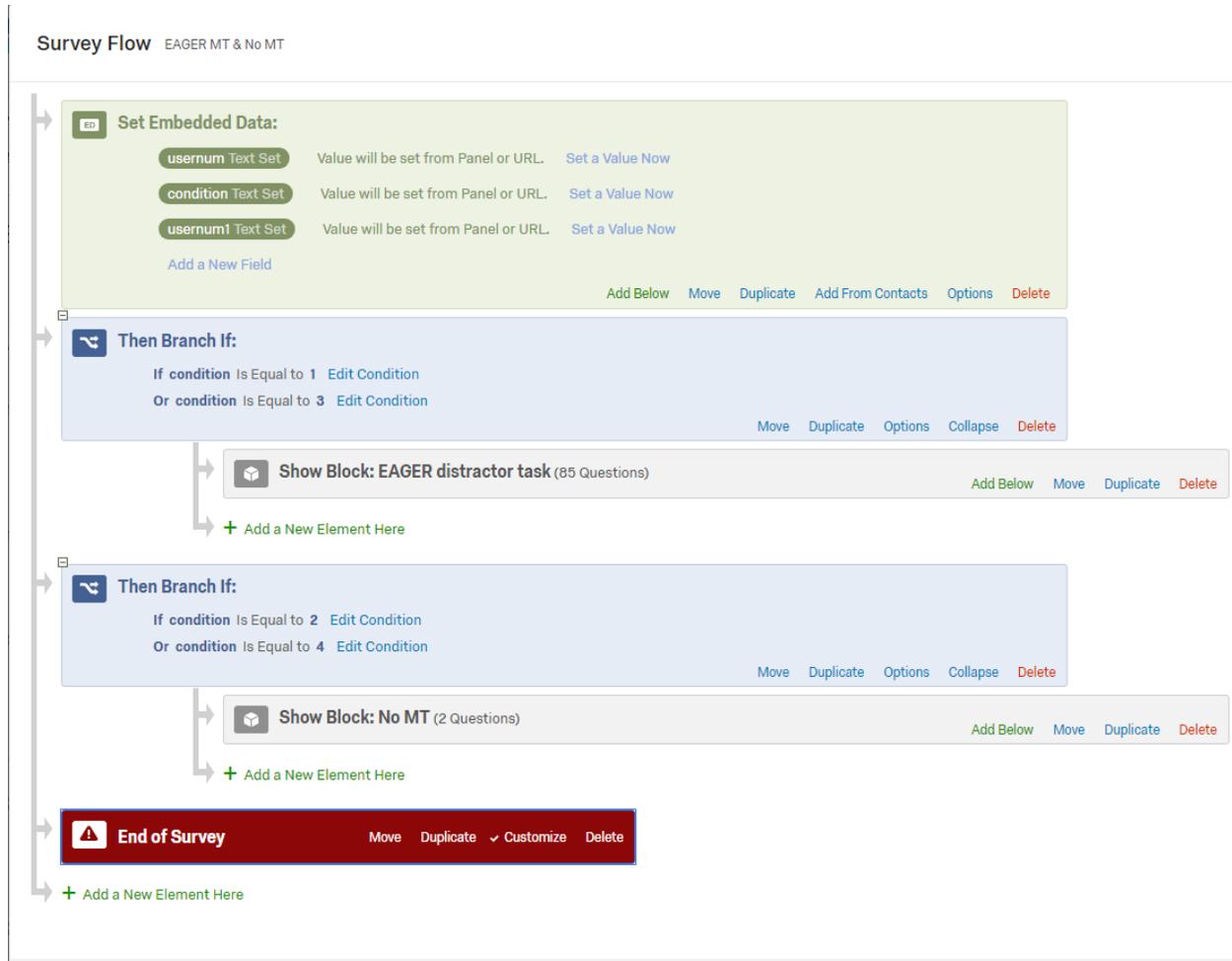
The screenshot displays a vertical sequence of three configuration steps in a survey flow:

- Step 1: Web Service**
 - URL: `http://reporting.qualtrics.com/projects/randomNumGen.php` (with a **Test URL** button)
 - Method: **GET**
 - Query Parameters:
 - `min` = `1`
 - `max` = `4`
 - Option: **Fire and Forget**
 - Set Embedded Data:
 - `condition` = `random`
 - Buttons: **Add Below**, **Move**, **Duplicate**, **Delete**
- Step 2: Set Embedded Data:**
 - Control: **MID Text Set**
 - Text: **Value will be set from Panel or URL.** (with **Set a Value Now** link)
 - Option: **Add a New Field**
 - Buttons: **Add Below**, **Move**, **Duplicate**, **Add From Contacts**, **Options**, **Delete**
- Step 3: Web Service**
 - URL: `http://ahab.isi.jhu.edu/service/service.php?condition=${e://Field/condition}` (with a **Test URL** button)
 - Method: **GET**
 - Options: **Add a query parameter to send to web service...** and **Add a custom header to send to web service...**
 - Option: **Fire and Forget**
 - Set Embedded Data:
 - `pid` = `pid`
 - `usern1` = `pid`
 - Buttons: **Add Below**, **Move**, **Duplicate**, **Delete**

The pre-survey continues and asks the participant to sign an agreement. The participant will be led to a series of cyber-related questions if they indicate their agreement. Otherwise, the pre-survey will be redirected to its end.



The participants will later be exposed to different sets of instructional questions based on different conditions they were previously assigned to. Finishing those instructional questions will lead them to several common questions and consequently the end of pre-survey.



At the end of the pre-survey, the page will be redirected to the following URL:

`http://ahab.isi.jhu.edu/roundcubemail-1.2.1/?condition=${e://Field/condition}`
`&usernum1=${e://Field/usernum1}`

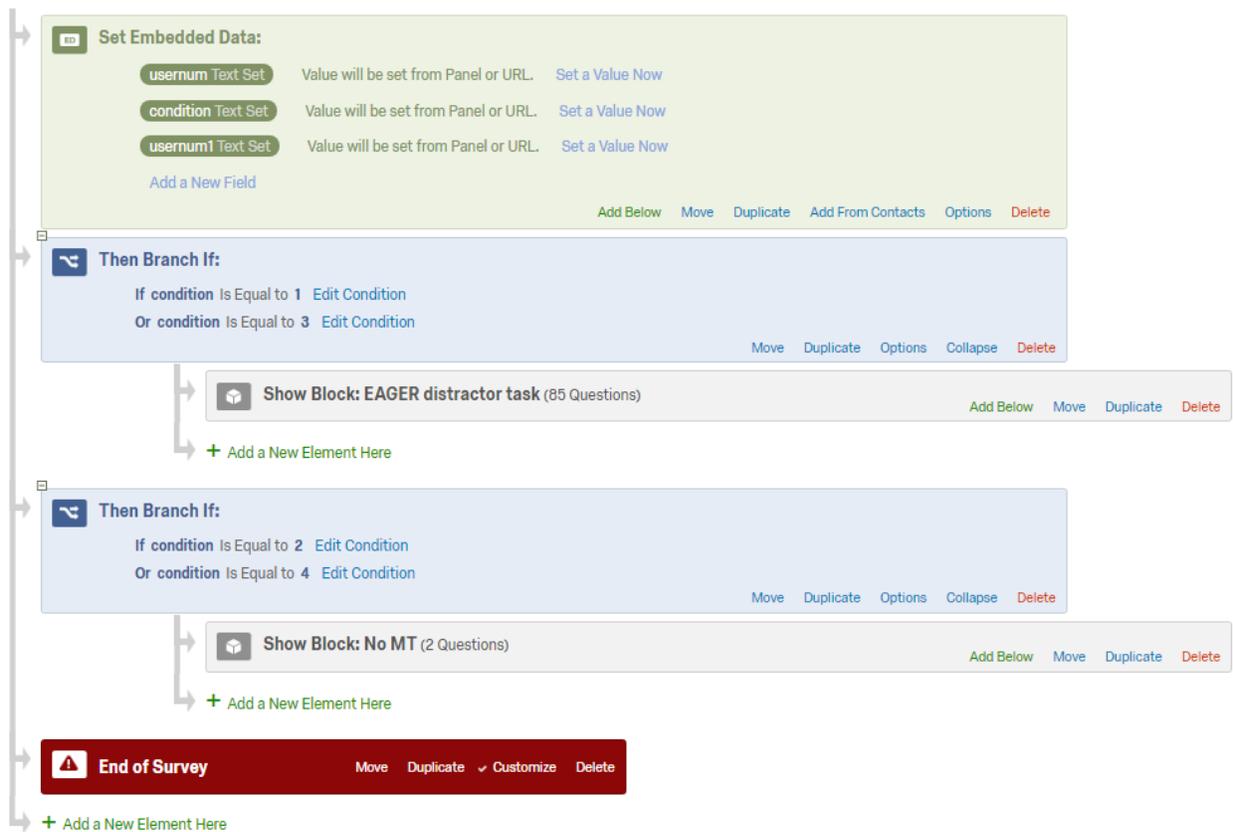
This URL take the participant to the combinational task of email sorting on roundcube and MT & NO MT survey with the condition field specified for different survey instructions and questions and usernum1 field specified for a dedicated email account.

Customize End Of Survey

- Override Survey Options**
- Default** end of survey message.
- Custom** end of survey message...
- Redirect to single response report.
- Redirect to a URL ...**
- Send additional thank you email from the library...
- Do not increment quota counts.
- Show Response Summary.
- Do NOT record any personal information and remove panel association (not recommended).
- Flag Response As
- Do NOT record survey response (not recommended).

b) Survey Flow of MT & NO MT Survey

The MT & NO MT Survey takes condition parameter to display according survey content while the usernum1 is passed through as one participant identifier.



[https://jhuapl.az1.qualtrics.com/jfe/form/SV_8D65Nunzx8bhlKB?condition=\\${e://Field/condition}&usenum=\\${e://Field/usenum}&usenum1=\\${e://Field/usenum1}](https://jhuapl.az1.qualtrics.com/jfe/form/SV_8D65Nunzx8bhlKB?condition=${e://Field/condition}&usenum=${e://Field/usenum}&usenum1=${e://Field/usenum1})

Customize End Of Survey

- Override Survey Options
 - Default** end of survey message.
 - Custom** end of survey message...
 - Redirect to single response report.
 - Redirect** to a URL ...
- Send additional thank you email from the library...
- Do not increment quota counts.
- Show Response Summary.
- Do NOT record any personal information and remove panel association (not recommended).
- Flag Response As
- Do NOT record survey response (not recommended).



c) Survey Flow of Post Survey

Upon completion of the MT & No-MT survey, Qualtrics would generate a random number (Conformation Number) ranged from 1000 to 9999 which will show up in the last page of the survey and be used as evidence of their complete participation of the survey. In this survey different set of questions will be asked based on different conditions.

The screenshot displays a survey flow editor with the following components:

- Web Service:** A configuration box for a REST API call. The URL is `http://reporting.qualtrics.com/projects/randomNumGen.php`. The method is `GET`. Query parameters include `min=1000` and `max=9999`. There is a checkbox for `Fire and Forget` and a `Set Embedded Data` section with a `random` field.
- Set Embedded Data:** A section for defining text sets. It includes:
 - `condition` Text Set = `#{e://Field/condition}`
 - `confirm` Text Set = `Se#{e://Field/random}*9199`
 - `usernum` Text Set: Value will be set from Panel or URL. [Set a Value Now](#)
 - `usernum1` Text Set: Value will be set from Panel or URL. [Set a Value Now](#)
- Show Block: email sorting (3 Questions):** A block with `Add Below`, `Move`, `Duplicate`, and `Delete` actions.
- Show Block: strategy (4 Questions):** A block with `Add Below`, `Move`, `Duplicate`, and `Delete` actions.
- Then Branch If:** A conditional branching step with the condition `If condition Is Equal to 1` and an `Edit Condition` link. It has `Move`, `Duplicate`, `Options`, `Collapse`, and `Delete` actions.
- Branch 1:** **Show Block: MT (4 Questions)** with `Add Below`, `Move`, `Duplicate`, and `Delete` actions.
- Branch 2:** **Show Block: incentive (3 Questions)** with `Add Below`, `Move`, `Duplicate`, and `Delete` actions.
- Footer:** `+ Add a New Element Here`

Then Branch If:
If condition Is Equal to 1 Edit Condition
Move Duplicate Options Collapse Delete

Show Block: MT (4 Questions)
Add Below Move Duplicate Delete

Show Block: incentive (3 Questions)
Add Below Move Duplicate Delete

+ Add a New Element Here

Then Branch If:
If condition Is Equal to 2 Edit Condition
Move Duplicate Options Collapse Delete

Show Block: incentive (3 Questions)
Add Below Move Duplicate Delete

+ Add a New Element Here

Then Branch If:
If condition Is Equal to 3 Edit Condition
Move Duplicate Options Collapse Delete

Show Block: MT (4 Questions)
Add Below Move Duplicate Delete

+ Add a New Element Here

Show Block: BFI (1 Question)
Add Below Move Duplicate Delete

End of Survey
Move Duplicate Customize Delete

+ Add a New Element Here

EAGER End of Experiment Message

Edit

Select Language ▾

All tasks and surveys are now complete. Thank you very much for your participation in our study!

Please enter the confirmation number below into the blank form in the HIT and then submit it. We cannot verify and pay you without this!

Confirmation Number: $\{e://Field/confirm\}$

As a reminder, you can ask questions about this research study, by talking to the researcher(s) by calling *Nathan Bos* at 443-778-1783, or email at *Nathan.bos@jhuapl.edu*.

If you have questions about your rights as a research participant or feel that you have not been treated fairly, please call the Homewood Institutional Review Board at Johns Hopkins University at (410) 516-6580.

Close

DESIGN & IMPLEMENTATION (CODE SNIPPET AND COMMENTS)

1. Roundcube Modifications

a) Disable Menu Buttons

```
var refresh = document.getElementsByClassName("button checkmail")[0];
var compose = document.getElementsByClassName("button compose")[0];
var reply = document.getElementsByClassName("button reply")[0];
var reply_all = document.getElementsByClassName("button reply-all")[0];
var forward = document.getElementsByClassName("button forward")[0];
var delete_mail = document.getElementsByClassName("button delete")[0];
var mark_mail = document.getElementsByClassName("button markmessage")[0];
var more_options = document.getElementsByClassName("button more")[0];
var logout = document.getElementsByClassName("button-logout")[0];

refresh.setAttribute("onclick", "");
refresh.setAttribute("href", "javascript: void(0)");

compose.setAttribute("onclick", "");
compose.setAttribute("href", "javascript: void(0)");

reply.setAttribute("onclick", "");
reply.setAttribute("href", "javascript: void(0)");

reply_all.setAttribute("onclick", "");
reply_all.setAttribute("href", "javascript: void(0)");

logout.setAttribute("onclick", "");
logout.setAttribute("href", "javascript: void(0)");
```

b) Disable Menu Buttons Dropdown

```
// Disabling the Reply-All DropDown
var replyAllDropDown = document.getElementById("replyallmenulink");
replyAllDropDown.addEventListener("click", function(){
    var replyAllTopDiv = document.getElementById("replyallmenu-menu");
    var replyAllOptions = replyAllTopDiv.childNodes
    for (i = 0; i < replyAllOptions.length; i++) {
        var option = replyAllOptions[i];
        if(i%2 != 0){
            option.childNodes[0].setAttribute("onclick", "");
            option.childNodes[0].setAttribute("href", "javascript:
void(0)");
        }
    }
});

forward.setAttribute("onclick", "");
forward.setAttribute("href", "javascript: void(0)");
```

c) Disable Right Click

```
// Disable Right Click
document.addEventListener('contextmenu', event => event.preventDefault());
```

d) Disable F5

```
// Disable F5 09-20-2017
function disableF5(e) { if ((e.which || e.keyCode) == 116) e.preventDefault(); };
/* jQuery < 1.7 */
$(document).bind("keydown", disableF5);
/* OR jQuery >= 1.7 */
$(document).on("keydown", disableF5);
```

2. Adding Event Listeners, Log Format and AJAX on Roundcube

a) Listeners on Qualtrics Iframe

```
// iFrame Mouse Listener
var d = new Date();
var timer = 0;

iframe.addEventListener("mouseover", function(){
    d = new Date();
    timer = d.getTime();
    sendLog(timer + " " + userId + " <Hover Over> Qualtrics")
});
iframe.addEventListener("mouseout", function(){
    d = new Date();
    timer = d.getTime() - timer
    if(timer > 100){
        sendLog(d.getTime() + " " + userId + " <Hover Out> of Qualtrics in
" + timer + " milliseconds");
    }
});

iframe.addEventListener('click', function() {
    var d = new Date();
    var timer = d.getTime()
    sendLog(d.getTime() + " " + userId + " <Clicked On> Qualtrics
Interface");
}, false);

// Mouse in-out of the window
$(document).mouseleave(function () {
    var d = new Date();
    var timer = d.getTime();
    sendLog(timer + " " + userId + " <Hover Out> of the Window");
});
$(document).mouseenter(function () {
    var d = new Date();
    var timer = d.getTime();
    sendLog(timer + " " + userId + " <Hover In> the Window");
```

```
});
```

b) Listeners on Folder Buttons

```
// Folder Tracker
var folders = document.getElementById("mailboxlist").childNodes
for (i = 0; i < folders.length; i++) {
    (function () {
        if(i%2 == 0){
            var link =
folders[i].childNodes[0].getAttribute("rel")

            folders[i].addEventListener('click', function() {
                d = new Date();
                timer = d.getTime();
                sendLog(timer + " " + userId + " <Clicked
On> Folder: " + link)
            }, false);
        }
    })();
}
```

c) AJAX Script to Send Event Capture Data

```
// AJAX Script to Send Event Capture Data
function sendLog(eventData){
    eventData = new Date().toString().match(/([A-Z]+[\+-][0-9]+.+)/)[1] + " "
+ eventData;
    console.log(eventData);
    $.ajax({
        url: "../Internal Test Logs/logger.php",
        type: "post",
        data: {'userId': userId, 'event': eventData} ,
        success: function (response) {
            //alert(response);
            //console.log("SENT mouse location:", e.clientX,
e.clientY)
        },
        error: function(jqXHR, textStatus, errorThrown) {
            console.log(textStatus, errorThrown);
        }
    });
}
```

d) Click Listener on Email List

```
// Mail Click Listeners
var userId = document.querySelector('.username').textContent;
d = new Date();
timer = d.getTime();
var el = document.createElement( 'html' );
el.innerHTML = this.innerHTML;
var subject = el.getElementsByTagName( 'a' )[0].innerText;
localStorage.setItem("sbjct",subject);
```

```

        var senderAddress = el.getElementsByClassName("adr")[0].innerText;
        //var date = el.getElementsByClassName("date")[0].innerText;
        sendLog(timer + " " + userId + " <Clicked On> Email: <" + subject
+ "> <" + senderAddress + ">");

```

e) Add “Move to” to Toolbar

```

// Add Move To in Toolbar
var moveTo = document.getElementById("messagetoolbar");
moveTo.innerHTML = moveTo.innerHTML + '<a class="button move" title="Move to..."
data-menu-pos="bottom" id="rcmbtn112" role="button" tabindex="0" aria-
disabled="false" href="#"
onclick="rcmail.command(\'move\',\'\',this,event);addListenersToMoveOptions();">Mo
ve</a>';

```

f) Add Listener to “Move to” Button

```

// Add Listeners to the Drop down List of the "Move To"
var moveOptionsFlag = false; // To ensure that the listeners are assigned only
once
function addListenersToMoveOptions(){

    if(moveOptionsFlag == false){
        var folderList = document.getElementById("folder-
selector").childNodes[0].childNodes;

        for (i = 0; i < folderList.length * 2; i++) {
            (function () {
                if(i%2 == 0){
                    var link =
folderList[i/2].childNodes[0].childNodes[0].innerHTML
                    folderList[i/2].addEventListener('click',
classification, false);
                }

                function classification(){
                    var iframe =
document.getElementById('messagecontframe');
                    var innerDoc = iframe.contentDocument ||
iframe.contentWindow.document;

                    if( innerDoc.getElementById("checkRating").innerHTML == 'n'){
                        createDialog("Alert", "You must
rate the confidence in your judgment before the email can be moved.");
                        return;
                    }
                    var d = new Date();
                    var timer = d.getTime();
                    var mail =
document.getElementsByClassName("message selected focused")[0];
                    var el =
document.createElement( 'html' );
                    el.innerHTML = mail.innerHTML;
                    var subject =
el.getElementsByTagName( 'a' )[0].innerText;

```

```

        var sbjct = localStorage.getItem("sbjct");
        //alert(sbjct);
        var senderAddress =
el.getElementsByClassName("adr")[0].innerText;
        //var date =
el.getElementsByClassName("date")[0].innerText;
        sendLog(timer + " " + userId + "
<Classified> Email: <" + sbjct + ">" + " to " + "<" + link + ">");
        //alertIfNotSelected = false;
    }
    }();
    }
    moveOptionsFlag = true;
}
}
}

```

g) Detect Mouse Move In or Out of Roundcube UI

```

// Mouse in-out of the window
$(document).mouseleave(function () {
    var d = new Date();
    var timer = d.getTime();
    sendLog(timer + " " + userId + " <Hover Out> of the Window");
});
$(document).mouseenter(function () {
    var d = new Date();
    var timer = d.getTime();
    sendLog(timer + " " + userId + " <Hover In> the Window");
});

```

h) Capture IP Address and Geo-Location

```

//Capture IP address - 9/15/17
$(document).ready(function () {
    $.getJSON("http://jsonip.com/?callback=?", function (data) {
        d = new Date();
        timer = d.getTime();
        sendLog(timer + " " + userId + " IP Address: " + data.ip);
    });
});

//Capture Geo-Location - 9/15/17
$.get("https://ipinfo.io", function(response) {
    d = new Date();
    timer = d.getTime();
    sendLog(timer + " " + userId + " Geo-location: " + response.city + " " +
response.country );
    //alert(response.city, response.country);
}, "jsonp");

```

i) Create New Alert UI

```

// New Alert UI
function createDialog(title, text) {

```

```

    return $("<div class='dialog' title='" + title + "'><p>" + text +
"</p></div>")
    .dialog({
        resizable: false,

        modal: true,
        buttons: {
            "Ok": function() {
                $( this ).dialog( "close" );
            }
        }
    });
}

```

j) Enable Alert When Email List Count Reaches Zero

```

// End Experiemnt
function endExperiment(){
    if(cdt == 3 || cdt == 1){
        createDialog('Alert', 'The email sorting task is complete.
Please continue with the distractor questions on the right side of the screen
until you are automatically redirected to the post-task questionnaire. Remember,
you will have to wait the full minute between each question.');
```

}

```

        else{
            createDialog('Alert', 'The email sorting task is complete.
You can click on the next button on the right side of the screen below the timer
to continue to the post-task questionnaire.')
```

}

```

        //window.location.href = "";
    }

/**
 * 'remove' message row from list (just hide it)
 */
remove_row: function(uid, sel_next)
{
    var self = this, node = this.rows[uid] ? this.rows[uid].obj : null;

    if (!node)
        return;

    node.style.display = 'none';

    // SK -----
    // Stop Roundcube to select the next row after move

    //if (sel_next)
    //    this.select_next();

    // SK -----

    delete this.rows[uid];
    this.rowcount--;
}

```

```

// SK -----
// Redirect when the Last mail has been sorted
if(this.rowcount == 0){
  //alert("Task Complete!");
  endExperiment();
}
// SK -----

// fix list header after removing any rows
clearTimeout(this.resize_timeout)
this.resize_timeout = setTimeout(function() { self.resize(); }, 50);
},

```

k) Browser Capability

```

// Browser Compatibility
var mailview = document.getElementById("mailview-right");
if((navigator.userAgent.indexOf("Opera") || navigator.userAgent.indexOf('OPR')) !=
-1 )
{
  // Do Nothing
}
else if(navigator.userAgent.indexOf("Chrome") != -1 )
{
  // Do Nothing
}
else if(navigator.userAgent.indexOf("Safari") != -1)
{
  mailview.style.width= "90%";
}
else if(navigator.userAgent.indexOf("Firefox") != -1 )
{
  mailview.style.width= "90%";
}
else if((navigator.userAgent.indexOf("MSIE") != -1 ) || (!!document.documentMode
== true )) //IF IE > 10
{
  mailview.style.width= "100%";}

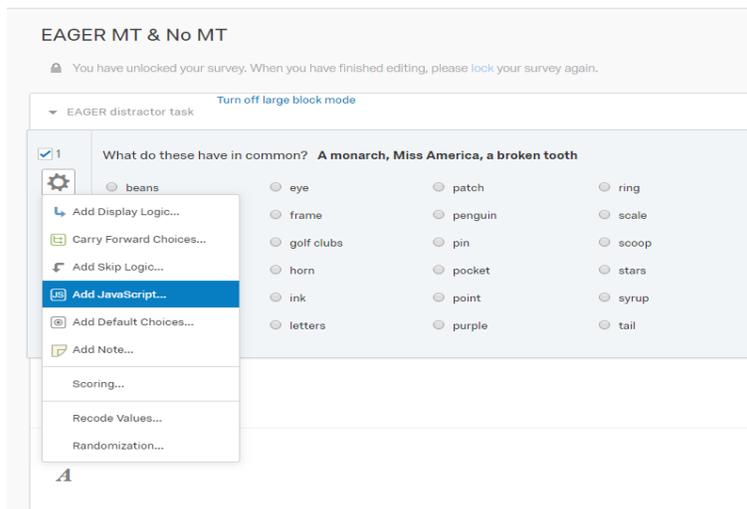
```

3. EVENT LISTENERS, LOG FORMAT AND AJAX ON QUALTRICS

Official Instructional Support

- Navigate to the **Survey** tab.

- Click the gray gear to the left of the question you want to add JavaScript to and select Add JavaScript.



- Place your JavaScript in the box, below one of the pieces of `/*Place Your JavaScript Here...*/` text. The opening and closing tags are already provided

Edit Question JavaScript

```

Qualtrics.SurveyEngine.addOnLoad(function()
{
  /*Place your JavaScript here to run when the page loads*/
});
Qualtrics.SurveyEngine.addOnReady(function()
{
  /*Place your JavaScript here to run when the page is fully displayed*/
});
Qualtrics.SurveyEngine.addOnUnload(function()
{
  /*Place your JavaScript here to run when the page is unloaded*/
});

```

3

JS Question API

4

- Click Save

a) Centered Timer

```

// Center Timer
Qualtrics.SurveyEngine.addOnload(function()
{
  var panel = document.getElementById("QID17Timer")
  var body = document.body;
  var w=panel.offsetWidth;
  panel.style.position="absolute";
  panel.style.top="90%";
  panel.style.left="50%";
  panel.style.marginLeft=- (w/2)+"px";
});

```

b) Get User and Question HTML Elements

```

// Get HTML Elements
var unum = document.getElementById("usernumber")

```

```

var uid = unum.innerHTML.replace(/^\s*|\s*$/g, "");
var currentQuestionID = this.getQuestionInfo().QuestionID
var questionobject = document.getElementById(currentQuestionID)
var
questionElement=questionobject.children[1].children[0].children[0].children[0].children[0]

```

c) Add Event Listeners to Question Body

```

//Add Event Listeners to Question Body
questionElement.addEventListener("mouseover", function(event,element){
    userId= "smithj"+uid+"@jhu.edu"
    d = new Date();
    timer = d.getTime();
    sendLog(timer + " " + userId + " <Hovered in> Question Body of
Question: "+ currentQuestionID)
});

questionElement.addEventListener("mouseout", function(event,element){
    userId= "smithj"+uid+"@jhu.edu"
    d = new Date();
    timer = d.getTime();
    sendLog(timer + " " + userId + " <Hovered in> Question Body of
Question: "+ currentQuestionID)
});

questionElement.addEventListener("click", function(event,element){
    userId= "smithj"+uid+"@jhu.edu"
    d = new Date();
    timer = d.getTime();
    sendLog(timer + " " + userId + " <Clicked on> Question Body of
Question: "+ currentQuestionID)
});

```

d) Get All the Choice Buttons HTML Elements

```

//Get ALL the Choice Buttons HTML Elements
var ChoiceButtons = [];
for (var i=1;i<25;i++)
{
    Choice = document.getElementById(currentQuestionID+"-"+i+"-label");
    console.log(Choice.textContent)//Test to See the text content of the
choice
    ChoiceButtons.push(Choice)
}

```

e) Add Event Listeners to Choice Buttons

```

//Add Event Listeners to Choice Buttons
for (var k=1;k<ChoiceButtons.length;k++){

ChoiceButtons[k].addEventListener("mouseover", (function(num){

    return function (e){
        userId= "smithj"+uid+"@jhu.edu"

```

```

        d = new Date();
        timer = d.getTime();
        num1 = num+1;
        sendLog(timer + " " + userId + " <Hovered in> Choice"+num1+" of
Question (" + currentQuestionID+")" );
    }
    })(k), 'false');

for (var k=1;k<ChoiceButtons.length;k++){

ChoiceButtons[k].addEventListener("mouseout", (function(num){

    return function (e){
        userId= "smithj"+uid+"@jhu.edu"
        d = new Date();
        timer = d.getTime();
        num1 = num+1;
        sendLog(timer + " " + userId + " <Hovered out> Choice"+num1+" of
Question (" + currentQuestionID+")" );
    }
    })(k), 'false');

for (var k=1;k<ChoiceButtons.length;k++){

ChoiceButtons[k].addEventListener("click", (function(num){

    return function (e){
        userId= "smithj"+uid+"@jhu.edu"
        d = new Date();
        timer = d.getTime();
        num1 = num+1;
        sendLog(timer + " " + userId + " <Clicked on> Choice"+num1+" of
Question (" + currentQuestionID+")" );
    }
    })(k), 'false');

```

f) Send Logged Data to Institute Server via AJAX

```

// Send Logged Data to Institute Server via AJAX
function sendLog(eventData){
    eventData = new Date().toString().match(/([A-Z][\+-][0-9]+.*)/)[1] + " " +
eventData;
    console.log(eventData);
    //console.log('-----');
    -----');
    var jq = jQuery.noConflict();
    jq.ajax({
        url: "https://ahab.isi.jhu.edu/Internal Test Logs/logger.php",
        type: "post",
        data: {'userId': userId, 'event': eventData} ,
        success: function (response) {
            //alert(response);

```

```

    //console.log("SENT mouse Location:", e.clientX, e.clientY)
  },
  error: function(jqXHR, textStatus, errorThrown) {
    console.log(textStatus, errorThrown);
  }
});
}

```

4. INTEGRATION BETWEEN ROUND CUBE AND QUALTRICS

a) Presentational Level

The iframe for Qualtrics Survey is embedded on the right side of the webpage as specified in the [mail.html](#).

```

<iframe id="iframe"
src="https://jhuapl.az1.qualtrics.com/jfe/form/SV_etYmSRRx50EvcBn"
style="width:30%; height:100%; left: 70%; position: fixed; margin-top: -
5%;"></iframe>

```

b) Logical Level

End of Survey Redirect to URL: [http://ahab.isi.jhu.edu/roundcubemail-](http://ahab.isi.jhu.edu/roundcubemail-1.2.1/?condition=${e://Field/condition}&usernum=${e://Field/usernum}&usernum1=${e://Field/usernum1})

[1.2.1/?condition=\\${e://Field/condition}&usernum=\\${e://Field/usernum}&usernum1=\\${e://Field/usernum1}](http://ahab.isi.jhu.edu/roundcubemail-1.2.1/?condition=${e://Field/condition}&usernum=${e://Field/usernum}&usernum1=${e://Field/usernum1})

Customize End Of Survey

- Override Survey Options**
- Default** end of survey message.
- Custom** end of survey message...
- Redirect to single response report.
- Redirect to a URL ...**
- Send additional thank you email from the library...
- Do not increment quota counts.
- Show Response Summary.
- Do NOT record any personal information and remove panel association (not recommended).
- Flag Response As
- Do NOT record survey response (not recommended).

[login.html](#)

```

var turkId = gup( 'usernum1' );
var condition = gup('condition')
var usernum1= gup('usernum1')
console.log(turkId);
console.log("condition: "+ condition)

localStorage.setItem("condition",condition);
localStorage.setItem("usernum1",usernum1);

// Set the Username Field
var userField = document.getElementById("rcmloginuser");
userField.value = turkId;

var task = gup( '_task' );
if(task != "logout"){
    var message = document.getElementById("message");
    message.innerHTML = "Logged Out!";
    document.forms[0].submit();
}

```

Hard-coded part of the user name with smithj which consist of part of the user name pattern, and the password used for all the accounts.

```

$auth = $RCMAIL->plugins->exec_hook('authenticate', array(
    'host' => $RCMAIL->autoselect_host(),
    'user' => trim(rcube_utils::get_input_value('_user',
rcube_utils::INPUT_POST)),
    'pass' => rcube_utils::get_input_value('_pass', rcube_utils::INPUT_POST,
true,
    $RCMAIL->config->get('password_charset', 'ISO-8859-1')),
    'cookiecheck' => true,
    'valid'      => $request_valid,
));

$auth['user'] = "newtest". $auth['user'] ."@jhu.edu";
// $auth['user'] = "smithj". $auth['user'] ."@jhu.edu";
$auth['pass'] = "password@123";

```

5. CONCUREENT USER ACCESS ISSUE

The issue: Depending on the Qualtrics end to control the numbering of the users would cause the concurrent user access issue. The counter on Qualtrics would not add until the participant has finished with the survey during which time a new participant may join in the survey and be assigned with the same condition and the same user account, which is being used by a previous user.

Solution:

Web Service

URL:
Test URL

Method: GET

[Add a query parameter to send to web service...](#)

[Add a custom header to send to web service...](#)

Fire and Forget

Set Embedded Data

=

-
+

=

-
+

[Add Below](#)
[Move](#)
[Duplicate](#)
[Delete](#)

```

<?php

$db = new SQLite3('rcuser_db') or die('Unable to open database');
$db->busyTimeout(2000);

$condition = $_GET['condition'];

if(!ctype_digit($condition)){
    echo "0";
    return;
}else if($condition == ""){
    echo "0";
    return;
}

$beginTransaction = $db->query('BEGIN TRANSACTION') or die('BEGIN TRANSACTION
Query failed');

    $result = $db->query('SELECT * FROM users WHERE condition = ' .
$condition. ' AND assigned = 0 ORDER BY rcnum ASC LIMIT 1') or die('Select Query
failed');
    if($row = $result->fetchArray()){
        $result = $db->query('UPDATE users SET assigned = 1 WHERE rcnum =
' . $row['rcnum']) or die('Update Query failed');
        echo "pid=" . $row['rcnum'];
    }else{
        echo "0";
    }

$endTransaction = $db->query('END TRANSACTION') or die('END TRANSACTION Query
failed');

?>

```

EXPERIMENT SETUP (SCRIPT SNIPPET AND COMMENTS)

1. USER ACCOUNTS CREATION

```
MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mail |
| mysql |
| performance_schema |
| roundcubemail |
+-----+
5 rows in set (0.00 sec)
```

Choose the database of the Postfix.

```
MariaDB [(none)]> use mail;
```

Insert users into the users table via following format:

```
MariaDB [mail]> INSERT INTO users(email,password) VALUES ('smithj4058@jhu.edu',
ENCRYPT('password',SUBSTRING(SHA('password@123'),-13)));
```

2. EMAILS POPULATING

The following piece of code is used to populated emails to multiple email accounts with the same naming pattern: smithj\$usernum@jhu.edu with parameters as specified in [emails.csv](#).

```
<?php
require_once "dSendMail2.inc.php";
// First of all: DISABLE TIME LIMIT and IGNORE USER ABORT! You really don't want
to be interrupted in the middle of this.
set_time_limit(0);
ignore_user_abort(true);
$m = new dSendMail2;
for($usernum=4046;$usernum<=4070;$usernum++){
$mailnum = 1;
if (($handle = fopen("emails.csv", "r")) !== FALSE) {
    while (($data = fgetcsv($handle, 1000, ",")) !== FALSE) {
        $file_name = $data[0];
        $from_addr = $data[1];
        $display_name = $data[2];
        $subject = $data[3];
        $m->setEMLFile($file_name);
        $m->setFrom( $from_addr, $display_name);
        $m->setTo("smithj$usernum@jhu.edu");
        $m->setSubject($subject);
        //sleep(5);
        $m->send();
        echo("Mail$mailnum sent for smithj $usernum\n");
        $mailnum++;
    }
}
```

```
    }  
    fclose($handle);  
}  
  
}
```

APPENDICES

Other non-technical decisions that are vital to the successful operation of this user study are revealed in the appendices.

- A. [QUALTRICS SURVEY QUESTIONS](#)
- B. [DATA DICTIONARY](#)
- C. [EXPERIMENT USE EMAILS DESIGN](#)

REFERENCES

<https://www.digitalocean.com/community/tutorials/how-to-install-and-configure-postfix-as-a-send-only-smtp-server-on-ubuntu-16-04>

<https://www.qualtrics.com/support/survey-platform/survey-module/question-options/add-javascript/>

<https://www.linuxtechi.com/install-latest-version-of-roundcube-centos-7/>