VB2020
localhost

30 September - 2 October, 2020 / vblocalhost.com

GROWTH AND COMMODITIZATION OF REMOTE ACCESS TROJANS

Veronica Valeros & Sebastian García
Czech Technical University in Prague, Czech Republic

veronica.valeros@aic.fel.cvut.cz
sebastian.garcia@agents.fel.cvut.cz

ABSTRACT

Remote access trojans (RATs) are an intrinsic part of traditional cybercriminal activities, and they have also become a standard tool in advanced espionage attacks and scams. There have been significant changes in the cybercrime world in terms of organization, attacks and tools in the last three decades, however, the overly specialized research on RATs has led to a seeming lack of understanding of how RATs in particular have evolved as a phenomenon. The lack of generalist research hinders the understanding and development of new techniques and methods to better detect them.

This work presents the first results of a long-term research project looking at remote access trojans. Through an extensive methodological process of collection of families of RATs, we are able to present an analysis of the growth of RATs in the last 30 years. Through a closer analysis of 11 selected RATs, we discuss how they have become a commodity in the last decade. Finally, through the collected information we attempt to characterize RATs, their victims, attacks and operators.

Preliminary results of our ongoing research have shown that the number of RATs has increased drastically in the past ten years and that nowadays RATs have become standardized commodity products that are not very different from each other.

INTRODUCTION

Remote access software is a type of computer program that allows an individual to have full remote control of the device on which the software is installed. In this research we distinguish between *remote access tool* and *remote access trojan*. A *remote access tool* refers to a type of remote access software used for benign purposes, such as *TeamViewer* [1] or *Ammyy Admin* [2], which are common tools used by billions of users worldwide. Remote access trojans, referred to in this paper as RATs, are a special type of remote access software where (i) the installation of the program is carried out without user consent, (ii) the remote control is carried out secretly, and (iii) the program hides itself in the system to avoid detection. The distinction between tools and trojans was created by defenders to make clear the difference between benign and malicious RATs, however in the underground, attackers claim all RATs are *remote access tools*.

There have been significant changes in the world of cybercrime during the last three decades. RATs are no exception. In the early days, RATs were developed for fun, to showcase skills, and to pull pranks. Developing your own RAT was an entry-level skill that inexperienced users were somehow expected to rapidly acquire. While the challenge of building highly functional RATs remains to today, their use has evolved. In the last decade more and more RATs have been openly commercialized and turned into standard tools for espionage, financial and state-sponsored attacks [3-5].

Although there are many reports on specific RAT family campaigns [6], there is no previous comprehensive research that looks at RATs as a whole. The growth and evolution of RATs appears to have escaped public attention so far. The lack of a more generalist research hinders the understanding and development of new techniques and methods to better detect them.

This paper aims to start a discussion on RATs as a unique phenomenon that requires further study. We argue that in the last decade a shift has occurred in the threat landscape, in which RATs have become a commodity.

In this paper we first present the most comprehensive timeline of the most well-known RATs in the last 30 years. With this information we analyse the growth of remote access trojans. We build on the collected information to describe the key technological elements of RATs, and characterize different aspects of RATs. Finally, we explore how RATs are commercialized, whether they have become a commodity, and share insights on the types of attacks and attackers using RATs.

The contributions of this paper are:

- The first and most comprehensive timeline of the last 30 years of RATs.
- An overview of the commoditization of the most well-known RATs in 2019-2020.
- A first analysis of the types of attacks and attackers using RATs.

METHODOLOGY

For this analysis we searched and methodologically compiled a comprehensive list of RATs starting from their first public appearance in 1996. The information was collected from public sources and inquiries within the community. This large collection is the basis of this work.

In order to study their specific characteristics, their users and how they are commercialized, we chose 11 RATs from those most common in marketplaces during the period 2019-2020: WebMonitor RAT, Android Voyager RAT, Remcos RAT, SpyNote RAT, Luminosity Link RAT, Omni Android RAT, Ozone RAT, Imminent Monitor RAT, NanoCore RAT, NetWire RAT and CyberGate RAT. These RATs were selected using the following methodology: first, we searched on well-known forums for RATs that users were talking about or recommending to each other. We consulted HackForums [7], Sinister.ly [8], and Nulled [9]. Second, we searched for these RATs on websites that were selling hacking tools,

software and other goods. Third, we limited the scope to those RATs that were being sold in two or more marketplaces. Finally, further information was collected from public intelligence sources such as blogs, news articles and forums on each of the selected RATs.

GROWTH OF REMOTE ACCESS TROJANS

In order to define a common ground to further the understanding of RATs, we first introduce some of the key technical elements of every remote access trojan. Then we present the results of our data collection in the form of a timeline, and discuss the growth of RATs over the last 30 years.

Key technical elements

The two key elements of any remote access trojan are the *client* and the *server*. Additional elements may include the *builder*, *plug-ins* and *crypter*. In this context, a server is the program installed on the victim's device, which is configured to connect back to the attacker. The client is the program used by the attacker to monitor and control infected victims: it allows the visualization of all active victim infections, displays general information about each infection, and allows individual actions to be performed manually on each victim.

The builder is a program used to create new RAT servers with different configurations. When attackers move infrastructure quickly, launch new attacks and require flexibility, builders save time and provide agility.

To add more capabilities to the existing RAT, some malware authors rely on plug-ins. This is not a widely used capability, however the most popular RATs support plug-ins. Good plug-ins are craved by the cybercrime community. These plug-ins are one of the main differentiators in terms of cost in the underground market.

To be more efficient and hard to detect, attackers use crypters to make the RAT servers fully undetectable (FUD). Crypters are programs that take a given program, read the code, encrypt it with a key, and automatically create a new program that contains the encrypted code and key to decrypt it. Upon execution the key will be used to automatically decrypt the original program. Crypters are used to avoid detection by anti-virus engines.

Timeline of remote access trojans

To better understand RATs as a phenomenon, we collected, investigated, and built a corpus of the most well-known RATs in history. We were able to find, reference and document many RATs since 1996 by looking at reports or code. These RATs were grouped into families, with slight variations of the same RATs grouped together. The final list contains 337 unique families of RATs, registering the first time seen, or date of the first public report about them. The collected information is visualized in the form of a timeline and shown in Figure 1.

The timeline is divided in three phases. The first phase is from 1990 to 1999; the second phase is from 2000 to 2009; and the third phase is from 2010 to 2019. Each phase is illustrated in Figure 1 by different types of lines. In Figure 1 we also highlight the 11 RATs that will be analysed in more detail in the following sections.

Increase in the number of families

From the information collected and shown in Figure 1, we can observe that the growth of RATs slowly increased during the first two decades, but it was not until 2010 that their growth spiked. The most prominent RATs we know nowadays were developed in the last decade. The sudden increase in the number of families is further illustrated in Figure 2, where we quantify the number of families per phase.

The oldest RAT was first developed in 1996 [10], however legitimate remote access tools were first created in 1989 [11]. Since then, the number of RATs has grown rapidly.

The first phase was marked by home-made RATs. In these years, everyone made their own RAT, however these did not prosper and were not heavily used. Among the most prominent were Back Orifice, Sub7 and Netbus, which together defined a generation by being innovative and disruptive.

The second phase, 2000-2009, showed a slight growth of more mature RATs that were intended for fun, but which started to be used for attacks and profits. Among the highlights of this period are Gh0st, PoisonIvy and DarkComet.

The third period, 2010-2019, showed an important shift. RATs became a commodity. The market matured, RAT sellers were expected to provide support, new features, and in some cases even to host part of the infrastructure.

There are many factors that influence such sudden changes like the one observed in this case, however this is not the focus of this paper.

CHARACTERISING REMOTE ACCESS TROJANS

As mentioned in the previous section, RATs have two key elements: the client and the server. In this section we characterize RATs in terms of functionality, software quality, and types of operators.

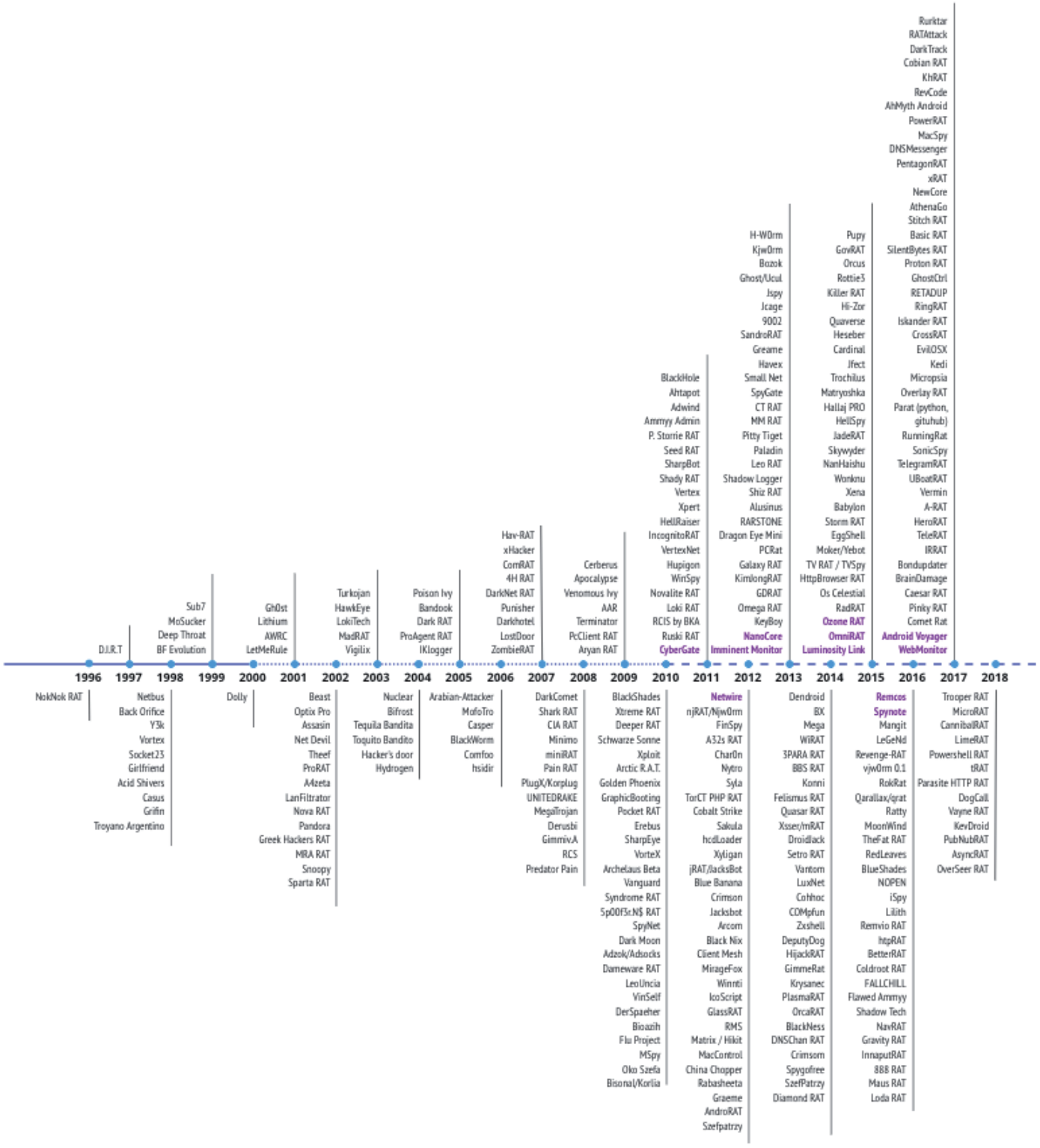

Figure 1: Timeline of 337 well-known remote access trojan families during 1996-2018. They are ordered by the year in which they were first seen or reported by the community. The last decade clearly shows a significant growth compared with the previous 16 years.

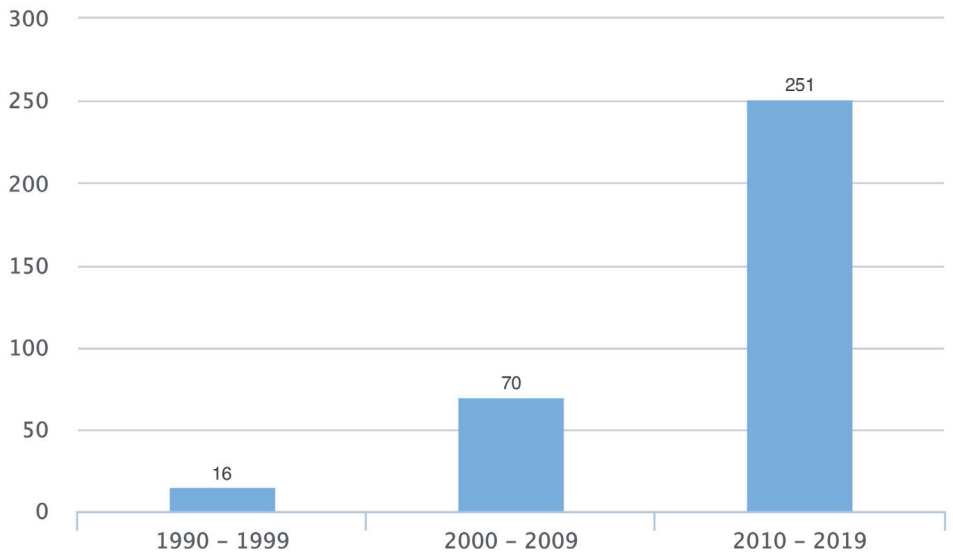

Figure 2: Number of RAT families per decade since 1990.

Functionality

In this work, functionality refers to what the software allows the operator to do on the victim side once the installation is successful. Although there is no standardized list of functionality, any RAT is expected to provide to a certain extent *access and control* over the following components:

- **Webcam**: take screenshots or full video recordings through the victim's webcam.
- **Microphone**: access the microphone to record audio.
- **Screen**: take screenshots or full screen recordings of the victim's desktop.
- **Keylogger**: capture keystrokes from the victim.
- **Operating system**: perform system operations such as retrieving system information, file management, hard drive and RAM access, installing programs, and other.
- **Peripherals**: access to peripheral devices including Bluetooth, CD/DVD reader, and others.

Software quality

Similarly to any other software, the success of a RAT not only depends on its functionality but also on its quality. The quality of a RAT can be measured by the same standards as any other software. ISO 9126 [12] defines that any high-quality software should have six characteristics: functionality, reliability, usability, efficiency, maintainability and portability. Each of these characteristics is described below in the context of RATs:

- **Functionality**: a high-quality RAT should ensure that the functionality offered, as described in the previous sub-section, is fully functional. This characteristic also includes security. In terms of RATs, encryption of traffic is a very important characteristic to keep the transferred information hidden in the network.
- **Reliability**: a high-quality RAT should ensure that the functionality works for at least a certain period of time without crashing and has certain tolerance to failures.
- **Usability**: a high-quality RAT should be easy to understand, to learn and to operate.
- **Efficiency**: a high-quality RAT should be efficient, making good use of the victim's resources. This is key for keeping the RAT undetected.
- **Maintainability**: a high-quality RAT is easy to update, to add features to, and stay functional while not sacrificing the usability and reliability.
- **Portability**: this characteristic refers to how the software can adapt to a change of environments. In terms of RATs it can refer to changes in Internet connection, or compatibility with other installed software and continuous use and operation of the normal user.

Roles: developers and operators

There are two clearly distinguished roles associated with every piece of malware: developers and operators. In many cases, these roles are carried out by the same actor. This was the case in the early days of RATs. Nowadays, however, these roles are associated with different actors, each with their own goals and purpose.

The developer(s) is in charge of creating the software, advertising and maintaining it. This actor has the coding skills necessary to develop the RAT and improve it to turn it into a high-quality software worth selling. This actor profits from selling the licences to use the software. In a simple scenario, this role also includes the advertising of the software and commercialization. However, these activities can easily be delegated to third parties.

The operator(s) is the actor who purchases the software (or a licence) and carries out the attacks. This actor has the knowledge of who the target is, the possible scams or attacks that can be carried out with the software, and which characteristics are needed when purchasing a RAT. This actor is also the one that pushes developers (and the market) for new functionalities, or new RATs altogether.

Commoditization of remote access trojans

Similarly to other types of malware, RATs are openly commercialized. This section provides insights into 11 of the most well-known RATs in 2019-2020, their characteristics, special features, and insights into their commercialization on different marketplaces.

Selected RATs

The selected RATs, as summarized in Table 1, are: WebMonitor RAT, Android Voyager RAT, Remcos RAT, SpyNote RAT, Luminosity Link RAT, Omni Android RAT, Ozone RAT, Imminent Monitor RAT, NanoCore RAT, NetWire RAT and CyberGate RAT.

RAT	First seen	Targeted platform	Used in targeted attacks	Client source code language	Server source code language
CyberGate RAT	2011	Windows	Yes	Delphi	C++
NetWire RAT	2012	Windows, Mac, Linux & Android	Yes	C	C
Imminent Monitor RAT	2012	Windows	Yes	.NET	.NET
NanoCore RAT	2013	Windows	Yes	.NET	.NET
Luminosity Link RAT	2015	Windows	Yes	.NET	.NET
Omni Android RAT	2015	Windows, Mac, Linux & Android	Yes	Java	Java
Ozone RAT	2015	Windows	Yes	C++	C++
Remcos RAT	2016	Windows	Yes	C++	C++
SpyNote RAT	2016	Android	Unknown	Visual Basic	Java
Android Voyager RAT	2017	Android	Unknown	Java	Java
WebMonitor RAT	2017	Windows, Mac, Linux & Google OS	Unknown	C++	C++

Table 1: Technical overview of 11 of the most common RATs during 2019-2020.

CyberGate RAT was first seen in 2011 [13]. The client is written in Delphi, the server is believed to be written in C++ and is very lightweight: 40KB uncompressed. It seems to share part of its code with an earlier RAT known as Xtreme RAT from 2010, whose source code was leaked [14]. This RAT specifically targets *Windows* machines, both 32-bit and 64-bit. Its key functionalities include keylogging, screenshot logging, password recovery and microphone capture.

NetWire RAT was first seen in 2012 [15] and is multi-platform, being able to target not only *Windows* machines but also *Mac*, *Linux* and *Android*. Both client and server are written in C. It offers full remote access with traditional functionalities such as keylogging, system management, password recovery and others. It allows heavy customization.

Imminent Monitor RAT, also known as IM-RAT, was first seen in 2012 [16]. It targets *Windows* machines, and both client and server are written in .NET. Imminent Monitor was commercialized as a benign administration tool, however researchers confirmed that some of its functionalities made the RAT undetectable for the victim, including recording from the webcam in an undercover manner by turning off the webcam light [17]. A version of Imminent Monitor allowed attackers to run a cryptocurrency miner on the infected machine.

NanoCore RAT was first publicly seen in 2013, while its author started coding it in late 2012 [18]. Both client and server are written in .NET. The source code of NanoCore has been leaked multiple times, and although the original author was

arrested there are versions of it still being sold today. NanoCore was reported to be used for state-sponsored attacks [3]. Among its features it offers a plug-in system to extend its functionality, remote chat, and uPnP support.

Luminosity Link was first seen in 2015 [19], targets *Windows* machines. Both client and server are written in .NET. The source code was leaked, and it is still being sold despite the fact its author was arrested [20].

Omni Android RAT, also known as OmniRAT, was first seen in 2015 [21]. This RAT is multi-platform, allowing it to target *Windows*, *Mac*, *Linux* and *Android* victims. For *Android* devices it allows a large amount of information to be retrieved, including battery level, widgets installed, Bluetooth, calls, and more.

Ozone-RAT was created in 2015 [22]. Both client and server are written in C++, and it targets specifically *Windows* machines. It offers traditional functionalities such as remote desktop, keylogging and system management. One of the main highlights of this RAT is that it offers a hidden VNC functionality.

Remcos RAT was first seen in 2016 [23]. Both client and server are written in C++, making it lightweight. Remcos targets 32-bit and 64-bit *Windows* machines. Its functionality includes uploading and downloading files, system management, and keylogging. There are several variants observed in the wild, which suggests that the source code may have been leaked.

SpyNote RAT version 2 was first seen in 2016 [24], however it may have been created earlier. The client is written in Visual Basic, and the server in Java. It targets *Android* devices. Among its functionality it includes the ability to access contacts, listen to calls, access front and back cameras, read SMS, and system management without requiring root access. The builder of SpyNote was leaked [25] and thus multiple versions of this RAT have been observed.

Android Voyager RAT, also known as Voyager RAT, was first seen in 2017 [26]. Both client and server are written in Java, and its author claims to be original and not based on any other leaked RAT. It targets *Android* devices. The functionality offered depends on whether there is root access on the device or not. Among the novel features, it claims that with root access it can survive factory reset on the *Android* device.

WebMonitor RAT was first seen in 2017 [27]. Both client and server are written in C++. WebMonitor targets *Windows*, *Linux*, *Mac* and *Google* OS. It's designed to be an enterprise class RAT able to compete with *TeamViewer* and other commercial remote access software. It offers stability, full remote control, and the management of clients through a web page being multi-platform on the client side as well.

Marketplaces

RATs are openly commercialized through forums and marketplaces. In this paper we focus on six marketplaces selling RATs among other hacking tools and services: DaVinciCoders, Secret Hacker Society, buyallrat588, Dorian Docs, FUD Exploits and Ultra Hacks. These are shown in Table 2, along with a summary of the RATs offered in each market and their prices.

RATs	**Sellers and marketplaces**					
	DaVinciCoders	Secret Hacker Society	buyallrat588	Dorian Docs	FUD Exploits	Ultra Hacks
CyberGate RAT	-	200	30-65	-	-	-
NetWire RAT	-	120	-	-	120	180
Imminent Monitor RAT	45	-	50-120	20-70	20-100	-
NanoCore RAT	45	96	-	-	150-170	-
Luminosity Link RAT	75	55	-	-	150	-
Omni Android RAT	-	80	60-150	120	120	180
Ozone RAT	75	-	-	-	170	-
Remcos RAT	-	99	-	-	170	-
SpyNote RAT	-	69	80-140	-	150-170	69
Android Voyager RAT	-	90	30-65	30-150	30	55-250
WebMonitor RAT	-	-	-	60-120	60	70-140

Table 2: Prices of commercialized RATs in online marketplaces.

DaVinciCoders (codevinci.pw) is a website that sells *Microsoft Office* exploits, crypters, keyloggers, RATs and botnets. It has four RATs on offer: Imminent Monitor, NanoCore, Luminosity Link and Ozone RAT. It offers plug-ins and support. The payment is handled via *rocketr.net*, however at the time of writing, the site has banned the products due to violations of their terms of services.

Secret Hacker Society (secrethackersociety.com) is a website that sells exploits, botnets, RATs, keyloggers, crypters, tutorials and hardware devices. It has nine RATs on offer, with prices ranging from 55 USD to 200 USD. Payments are handled via *perfectmoney.is* or Bitcoin.

Buy All Rat (buyallrat588.com) is a website that sells hacking software, RATs, exploits, spoofers, private mailers, SMTP, botnets, crypters, shells, VPNs, keyloggers and others. It has five RATs on offer on two tiers: basic and pro. Pro licences are more expensive as they typically include lifetime access, technical support, a one-week money back guarantee, and more than one device licensing. The seller doesn't perform direct sales, customers need to send an email request and all exchange is done privately.

Dorian Docs (doriandocs.com) is a website that sells accounts, RATs, fake IDs and fake documents. It has four RATs on offer and each RAT has a different tier: single price, business/professional, startup/small business/business, one month/three months/six months/lifetime. Payments are made with cryptocurrency, with Bitcoin, Monero, Ethereum and Litecoin all accepted.

FUD Exploits (fudexploits.com) is a website that sells botnets, crypters, passports, RATs and other products. It has ten RATs on offer, and it offers different RAT packages at different prices, varying according to versions, number of plug-ins, and support. Payments are made using Bitcoin.

Ultra Hacks (ultrahacks.org) is a website that sells tutorials, RATs, botnets, hardware, and services. It has five RATs on offer, only two of which are offered in two tiers, professional/premium, while the rest are offered in a single option. The seller accepts payments in Bitcoin, Monero, Litecoin, direct by transfer SEPA, cash on delivery and Perfect Money.

Commoditized product

The analysis of the market suggests that, far from being custom-made unique tools, RATs have become a commodity. They have become a group of standardized products that are not very different from each other. The variation in prices is not determined by the functionality of the RATs per se, but instead by the sellers themselves being able to offer additional services, extended functionality or technical support. No matter the skill level, attackers are able to choose from a wide range of very affordable options and adjust their attack to the final product selected. The most successful RATs do not have a huge technological advantage, but better reviews, recommendations and, in the end, better marketing.

CHARACTERIZING ATTACKS AND ATTACKERS

To better understand the market and context of these RATs, this section provides a first analysis of different known types of attacks performed with RATs and the different sectors or types of crimes they focus on. We also provide a characterization of the attackers/operators behind these attacks.

Attacks

In contrast with botnets, RATs are precision tools that excel in targeted attacks intended to extract specific information from victims. RAT attacks differ from most malware attacks in several aspects. First, contrary to botnets where an attacker controls all the bots simultaneously, attackers control each RAT infection individually and manually. Second, due to this individual control of each victim, the sequence of actions on the victims may never be the same in any two infections. Third, the number of simultaneous infections that an attacker can control is limited by the skill of the attacker. No attacker will be able to control half a million victims as they can do with botnets.

The most common types of attacks that use RATs as a central tool are:

- **Business email compromise (BEC):** a type of scam directed at companies or organizations that pay their suppliers via wire transfers [28]. Attackers aim to redirect the transfer of funds to attackers' accounts instead of the legitimate ones, thus stealing the money. While traditionally information stealers were the preferred tool in BEC attacks, there has been a shift and nowadays the use of RATs is becoming the norm [29].
- **Cyber espionage:** the act of stealing confidential information using software tools, such as malware. RATs excel in cyber espionage attacks. Cyber espionage attackers may develop their own RATs [30-32], or use well-known commercial RATs for a highly confidential operation. There are pros and cons for each case. Custom developed RATs may leave traces that can lead to the identification of the attacker, while commercial RATs may be better at hiding the operator's origin and intent. However, custom developed RATs may provide more stability, stealthiness and functionality than a commercial RAT could ever provide.
- **Targeted attacks:** attacks that are carefully planned, target a very narrow set of victims, and have often a very specific goal. RATs are widely used in this type of attack. Of the 11 RATs mentioned in the previous section, nine of them were used in targeted attacks [33-41].

Attackers

RAT attackers/operators are not homogeneous. They can be separated in three groups according to how the RAT is used: (i) for educational purposes, fun or pranks, (ii) for advanced attacks and espionage activities, and (iii) for cybercrime (whether selling RATs to other actors or buying RATs for attacking). We describe each of these groups:

- **Educational purposes**: attackers that use RATs to learn, for fun or pranks. These actors rarely purchase commercial RATs. They will write their own or modify existing ones. The renowned development platform *GitHub* [42] contains dozens or even hundreds of self-made RATs created and shared publicly with the disclaimer of being for education purposes only. In underground forums, the hacker community still believes that real hackers will create their own RAT, which incentivizes this activity.
- **Advanced attacks**: state-sponsored attackers and cybercrime organizations that conduct complex scams or targeted attacks. These actors are believed to create their own tools customized to their own needs. The use of open-source tools, however, may be useful in some scenarios to give false flags or as a distraction.
- **Cybercrime**: traditional cybercrime groups that conduct simple scams, stealing credit cards, or extortion attacks. These actors are the ones engaged in commercializing RATs. Sellers will use available RATs, modify or enhance them, package them and sell them. They will offer technical support, tutorials, and hosting services. Buyers do not want to get absorbed in technical details and programming, they look to focus on the attacks. Buyers rely on sellers to provide stable tools, with support and the ability to develop further modules for them in case they need them.

FUTURE WORK

Malicious software is evolving fast, many times mixing functionality that impedes the creation of a clear malware classification. In this paper we discuss remote access trojans, and we do not include spyware or stalkerware. Spyware and stalkerware, both intended to extract information from the device they are installed on and control the peripherals (GPS, camera, etc.), share many similarities with RATs, however they are addressed as separate threats by the community. Our future work will address these distinctions in detail.

CONCLUSION

In this work, we presented the first results of a long-term collection of information on remote access trojans. The timeline of the 337 most well-known RAT families from 1996-2018 allowed us to have a better understanding of the growth of RATs as a unique phenomenon. Insights into 11 of the most prominent RATs during 2019-2020 regarding their commercialization in online marketplaces showed that RATs are technologically not so different from each other. The main differences are in price due to added features or additional services offered by the sellers themselves. While it is still believed that real hackers will create their own RATs, business-oriented cybercriminals will look for stability, simplicity, support and guarantee; thus buying RATs instead of crafting their own. These commercial characteristics of RATs mark them as a commodity. Shifts in cybercriminal activities continue to happen and RATs are used more and more in all types of attacks. Their continual growth challenges current detection methods and asks for further research that focuses on RATs as a general malware class and not only in individual RAT families.

ACKNOWLEDGEMENTS

The authors would like to thank the Czech Technical University for its support. The authors would also like to thank all the individual researchers that shared information and helped during the last three years in building the RAT timeline.

REFERENCES

[1] TeamViewer: remote access, remote control and remote support solution, TeamViewer Germany GmbH. https://www.teamviewer.com/.

[2] Ammyy Admin: Remote Desktop Software and Remote Desktop Connection, Ammyy, Inc. http://www.ammyy.com/.

[3] Kovacs, E. Nation-State Actors Use Fileless Tricks to Deliver RATs. 2016. https://www.securityweek.com/nation-state-actors-use-fileless-tricks-deliver-rats.

[4] Marczak, W. R.; Scott-Railton, J.; Marquis-Boire, M.; Paxson,V. When governments hack opponents: A look at actors and technology. Proceedings of the 23rd USENIX Security Symposium, pp.511–525, 2014.

[5] Higgins, K. J. Schneider Electric: TRITON/TRISIS Attack Used 0-Day Flaw in its Safety Controller System, and a RAT. 2018. https://www.darkreading.com/vulnerabilities---threats/schneider-electric-triton-trisis-attack-used-0-day-flaw-in-its-safety-controller-system-and-a-rat/d/d-id/1330845.

[6] Rezaeirad, M.; Farinholt, B.; Dharmdasani, H.; Pearce, P.; Levchenko, K.; McCoy, D. Schrödinger's RAT: Profiling the stakeholders in the remote access trojan ecosystem. In 27th USENIX Security Symposium (USENIX Security

18). Baltimore, MD: USENIX Association, Aug. 2018, pp. 1043–1060. https://www.usenix.org/conference/usenixsecurity18/presentation/rezaeirad.

[7] HackForums. Accessed 5 March 2020. https://www.hackforums.net/.

[8] Sinisterly Forum. Accessed 5 March 2020. https://sinister.ly/.

[9] Nulled Forum. Accessed 5 March 2020. https://www.nulled.to/.

[10] MegaSecurity, NokNok 5.0. Accessed via Internet Archive. https://web.archive.org/web/20081201090344/http://www.megasecurity.org/trojans/n/noknok/Noknok5.0.html.

[11] House, N. NetSupport Manager – Multi-Platform Remote Control software. Accessed 6 March 2020. http://www.netsupportmanager.com/.

[12] ISO 9126 Software Quality Characteristics. Accessed 15 June 2020. http://www.sqa.net/iso9126.html.

[13] Ali, W. Cybergate rat – hacking facebook, twitter and email id's passwords. Accessed 6 March 2020. http://www.hackersthirst.com/2011/03/cybergate-rat-hacking-facebook-twitter.html.

[14] Villeneuve, N.; Bennett, J. T. Xtremerat: Nuisance or threat? Accessed 6 March 2020. https://www.fireeye.com/blog/threat-research/2014/02/xtremerat-nuisance-or-threat.html.

[15] MalwareMustDie. Mmd-0031-2015 – what is netwire (multiplatform) rat? Accessed 6 March 2020. https://blog.malwaremustdie.org/2015/04/mmd-0031-2015-what-is-netwire-rat.html.

[16] Seals, T. Authorities break up imminent monitor spyware organization. Accessed 6 March 2020. https://threatpost.com/authorities-imminent-monitor-spyware-organization/150731/.

[17] Palo Alto Unit42. Imminent monitor – a rat down under. Accessed 6 March 2020. https://unit42.paloaltonetworks.com/imminent-monitor-a-rat-down-under/.

[18] Poulsen, K. FBI arrests hacker who hacked no one. Accessed 6 March 2020. https://www.thedailybeast.com/fbi-arrests-hacker-who-hacked-no-one.

[19] Grunzweig, J. Investigating the luminositylink remote access trojan configuration. Accessed 6 March 2020. https://unit42.paloaltonetworks.com/unit42-investigating-the-luminositylink-remote-access-trojan-configuration/.

[20] Krebs, B. 'luminositylink rat' author pleads guilty. Accessed 6 March 2020. https://krebsonsecurity.com/2018/07/luminositylink-rat-author-pleads-guilty/.

[21] Chrysaidos, N. Droidjack isn't the only spying software out there: Avast discovers omnirat. Accessed 6 March 2020. https://blog.avast.com/2015/11/05/droidjack-isnt-the-only-spying-software-out-there-avast-discovers-that-omnirat-is-currently-being-used-and-spread-by-criminals-to-gain-full-remote-co.

[22] Honest ozone rat review. Accessed 6 March 2020. https://raidforums.com/Thread-Honest-Ozone-Rat-Review.

[23] Hinchliffe, A. Emea bi-monthly threat reports: Turkey, saudi arabia & united arab emirates. Accessed 6 March 2020. https://unit42.paloaltonetworks.com/unit42-emea-bi-monthly-threat-reports-turkey-saudi-arabia-united-arab-emirates/.

[24] Spynote [android rat] v2.3 server setting. Accessed 7 March 2020. https://www.youtube.com/watch?v=voeLG1H6qSY.

[25] Soo, J. Spynote android trojan builder leaked. Accessed 7 March 2020. https://unit42.paloaltonetworks.com/unit42-spynote-android-trojan-builder-leaked/.

[26] Voyager rat. Accessed 6 March 2020. https://hackforums.net/showthread.php?tid=5723439.

[27] Webmonitor pc [#1 rat on the market, c++/native (no .net), no portforward, keylogger]. Accessed 6 March 2020. https://hackforums.net/showthread.php?tid=5621975&highlight=Webmonitor.

[28] Business email compromise (bec). Accessed 7 March 2020. https://www.trendmicro.com/vinfo/us/security/definition/business-email-compromise-(bec).

[29] Ilascu, I. Nigerian BEC Scammers Shifting to RATs As Tool of Choice. Accessed 7 March 2020. https://www.bleepingcomputer.com/news/security/nigerian-bec-scammers-shifting-to-rats-as-tool-of-choice/.

[30] Zykov, K. Hello! My name is Dtrack. Accessed 7 March 2020. https://securelist.com/my-name-is-dtrack/93338/.

[31] Miller-Osborn, J.; Harbison,M. Rancor: Cyber Espionage Group Uses New Custom Malware to Attack Southeast Asia. Accessed 7 March 2020. https://unit42.paloaltonetworks.com/rancor-cyber-espionage-group-uses-new-custom-malware-to-attack-southeast-asia/.

[32] Rascagneres, P.; Mercer, W.; Ventura, V. Bisonal: 10 years of play. Accessed 7 March 2020. https://blog.talosintelligence.com/2020/03/bisonal-10-years-of-play.html.

[33] Mimoso, M. AutoIt Used in Targeted Attacks to Move RATs. Accessed 7 March 2020. https://threatpost.com/autoit-used-in-targeted-attacks-to-move-rats/114406/4/.

[34] Netwire RAT Behind Recent Targeted Attacks. Accessed 7 March 2020. https://www.kashifali.ca/2015/03/02/netwire-rat-behind-recent-targeted-attacks/.

[35] Cimpanu,C. Authorities take down 'Imminent Monitor' RAT malware operation. Accessed 7 March 2020. https://www.zdnet.com/article/authorities-take-down-imminent-monitor-rat-malware-operation/.

[36] Baezner, M. Regional rivalry between India-Pakistan: tit-for-tat in cyberspace. Accessed 7 March 2020. https://css.ethz.ch/content/dam/ethz/special-interest/gess/cis/center-for-securities-studies/pdfs/Cyber-Reports-2018-04.pdf.

[37] Kumar, M. Exclusive: German Police Raid OmniRAT Developer and Seize Digital Assets. Accessed 7 March 2020. https://thehackernews.com/2019/06/police-raid-omnirat-developer.html

[38] Bacurio, F. Jr.; Salvio, J. German Speakers Targeted by SPAM Leading to Ozone RAT. Accessed 7 March 2020. https://www.fortinet.com/blog/threat-research/german-speakers-targeted-by-spam-leading-to-ozone-rat.html.

[39] Remcos RAT Abuses Office Vulnerabilities to Target Businesses. Accessed 7 March 2020. https://www.enigmasoftware.com/remcos-rat-abuses-office-vulnerabilities-target-businesses/.

[40] Abel, R. Spynote RAT posing as Netflix plus other popular apps. Accessed 7 March 7, 2020. https://www.scmagazine.com/home/security-news/cybercrime/spynote-rat-posing-as-netflix-plus-other-popular-apps/.

[41] Desai, S. SpyNote RAT posing as Netflix app. Accessed 7 March 2020. https://www.zscaler.com/blogs/research/spynote-rat-posing-netflix-app.

[42] GitHub development platform. https://github.com/.