Jun 28 2024

NETMANAGEIT

Intelligence Report

An Android RAT targets Telegram Users

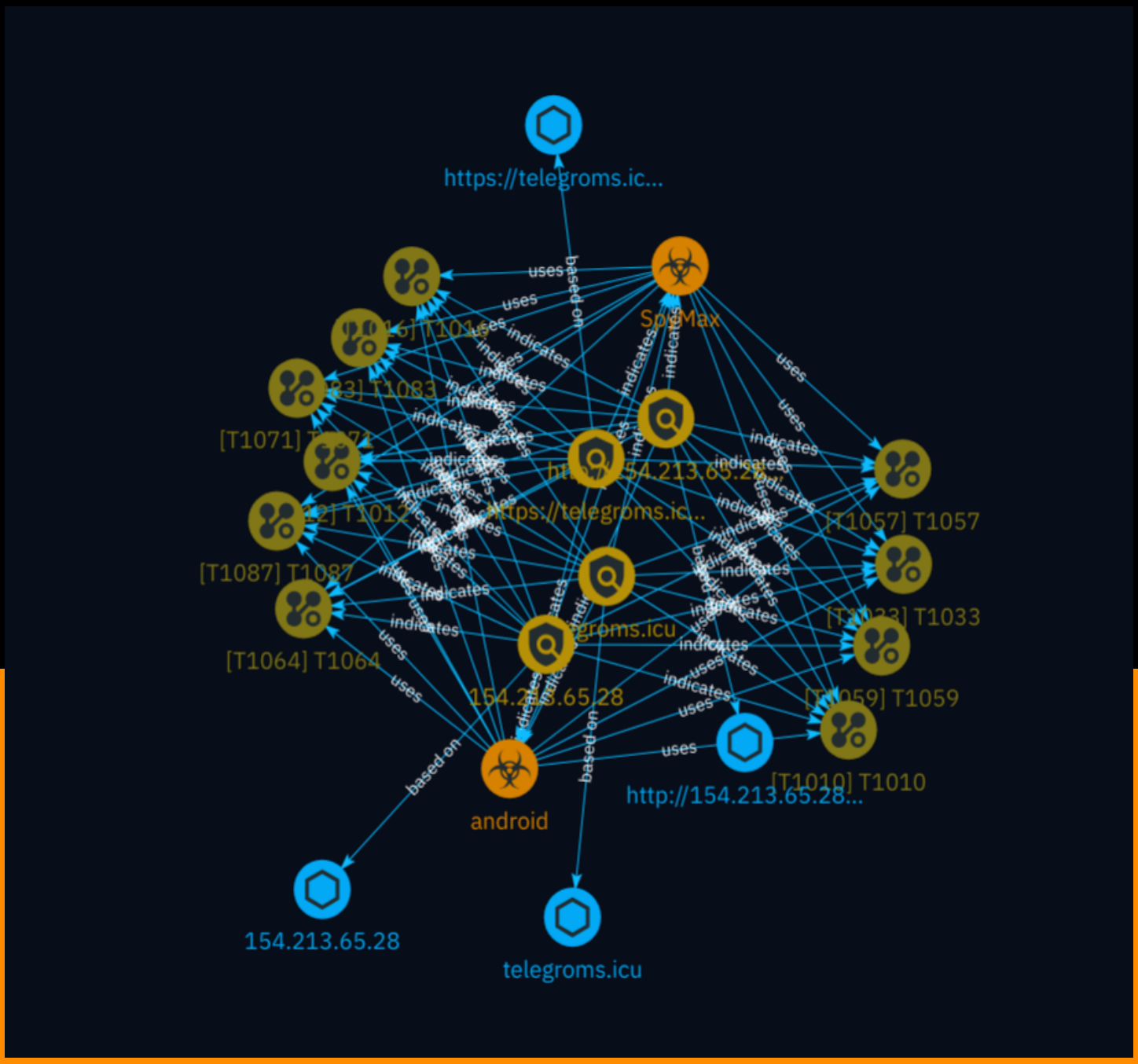

Table of contents

Overview

Entities

Observables

External References

Overview

Description

This analysis discusses SpyMax, a Remote Access Trojan (RAT) that targets Android devices and specifically aims at obtaining data from Telegram users. It employs phishing techniques to trick victims into installing a malicious application disguised as the legitimate Telegram app. Once installed, SpyMax gains extensive permissions, gathers sensitive information like keystrokes and location data, and transmits it to a remote command-and-control server. The malware also receives commands and additional payloads from the server, enabling remote control of the compromised device. The report outlines the technical details of SpyMax's operations, including its obfuscation methods, data exfiltration process, and communication with the command-and-control infrastructure.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

100 / 100

Content

N/A

Attack-Pattern

Name
T1057

ID
T1057

Description
Adversaries may attempt to get information about running processes on a system. Information obtained could be used to gain an understanding of common software/applications running on systems within the network. Administrator or otherwise elevated access may provide better process details. Adversaries may use the information from [Process Discovery](https://attack.mitre.org/techniques/T1057) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. In Windows environments, adversaries could obtain details on running processes using the [Tasklist](https://attack.mitre.org/software/S0057) utility via [cmd](https://attack.mitre.org/software/S0106) or `Get-Process` via [PowerShell](https://attack.mitre.org/techniques/T1059/001). Information about processes can also be extracted from the output of [Native API](https://attack.mitre.org/techniques/T1106) calls such as `CreateToolhelp32Snapshot`. In Mac and Linux, this is accomplished with the `ps` command. Adversaries may also opt to enumerate processes via `/proc`. On network devices, [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands such as `show processes` can be used to display current running processes. (Citation: US-CERT-TA18-106A)(Citation: show_processes_cisco_cmd)

Name
T1064

ID

T1064

Description

This technique has been deprecated. Please use [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059) where appropriate. Adversaries may use scripts to aid in operations and perform multiple actions that would otherwise be manual. Scripting is useful for speeding up operational tasks and reducing the time required to gain access to critical resources. Some scripting languages may be used to bypass process monitoring mechanisms by directly interacting with the operating system at an API level instead of calling other programs. Common scripting languages for Windows include VBScript and [PowerShell](https://attack.mitre.org/techniques/T1086) but could also be in the form of command-line batch scripts. Scripts can be embedded inside Office documents as macros that can be set to execute when files used in [Spearphishing Attachment](https://attack.mitre.org/techniques/T1193) and other types of spearphishing are opened. Malicious embedded macros are an alternative means of execution than software exploitation through [Exploitation for Client Execution](https://attack.mitre.org/techniques/T1203), where adversaries will rely on macros being allowed or that the user will accept to activate them. Many popular offensive frameworks exist which use forms of scripting for security testers and adversaries alike. Metasploit (Citation: Metasploit_Ref), Veil (Citation: Veil_Ref), and PowerSploit (Citation: Powersploit) are three examples that are popular among penetration testers for exploit and post-compromise operations and include many features for evading defenses. Some adversaries are known to use PowerShell. (Citation: Alperovitch 2014)

Name

T1012

ID

T1012

Description

Adversaries may interact with the Windows Registry to gather information about the system, configuration, and installed software. The Registry contains a significant amount of

information about the operating system, configuration, software, and security.(Citation: Wikipedia Windows Registry) Information can easily be queried using the [Reg](https://attack.mitre.org/software/S0075) utility, though other means to access the Registry exist. Some of the information may help adversaries to further their operation within a network. Adversaries may use the information from [Query Registry](https://attack.mitre.org/techniques/T1012) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions.

Name

T1010

ID

T1010

Description

Adversaries may attempt to get a listing of open application windows. Window listings could convey information about how the system is used.(Citation: Prevailion DarkWatchman 2021) For example, information about application windows could be used identify potential data to collect as well as identifying security tooling ([Security Software Discovery](https://attack.mitre.org/techniques/T1518/001)) to evade.(Citation: ESET Grandoreiro April 2020) Adversaries typically abuse system features for this type of enumeration. For example, they may gather information through native system features such as [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059) commands and [Native API](https://attack.mitre.org/techniques/T1106) functions.

Name

T1016

ID

T1016

Description

Adversaries may look for details about the network configuration and settings, such as IP and/or MAC addresses, of systems they access or through information discovery of remote systems. Several operating system administration utilities exist that can be used to gather this information. Examples include [Arp](https://attack.mitre.org/software/S0099), [ipconfig](https://attack.mitre.org/software/S0100)/[ifconfig](https://attack.mitre.org/software/S0101), [nbtstat](https://attack.mitre.org/software/S0102), and [route](https://attack.mitre.org/software/S0103). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather information about configurations and settings, such as IP addresses of configured interfaces and static/dynamic routes (e.g. `show ip route`, `show ip interface`).(Citation: US-CERT-TA18-106A)(Citation: Mandiant APT41 Global Intrusion) Adversaries may use the information from [System Network Configuration Discovery](https://attack.mitre.org/techniques/T1016) during automated discovery to shape follow-on behaviors, including determining certain access within the target network and what actions to do next.

Name

T1059

ID

T1059

Description

Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. These interfaces and languages provide ways of interacting with computer systems and are a common feature across many different platforms. Most systems come with some built-in command-line interface and scripting capabilities, for example, macOS and Linux distributions include some flavor of [Unix Shell](https://attack.mitre.org/techniques/T1059/004) while Windows installations include the [Windows Command Shell](https://attack.mitre.org/techniques/T1059/003) and [PowerShell](https://attack.mitre.org/techniques/T1059/001). There are also cross-platform interpreters such as [Python](https://attack.mitre.org/techniques/T1059/006), as well as those commonly associated with client applications such as [JavaScript](https://attack.mitre.org/techniques/T1059/007) and [Visual Basic](https://attack.mitre.org/techniques/T1059/005). Adversaries may abuse these technologies in various ways as a means of executing arbitrary commands. Commands and scripts can be embedded in [Initial Access](https://attack.mitre.org/tactics/TA0001) payloads delivered to victims as lure documents or as secondary payloads downloaded from an existing C2. Adversaries may also execute

commands through interactive terminals/shells, as well as utilize various [Remote Services](https://attack.mitre.org/techniques/T1021) in order to achieve remote Execution. (Citation: Powershell Remote Commands)(Citation: Cisco IOS Software Integrity Assurance - Command History)(Citation: Remote Shell Execution in Python)

Name

T1087

ID

T1087

Description

Adversaries may attempt to get a listing of valid accounts, usernames, or email addresses on a system or within a compromised environment. This information can help adversaries determine which accounts exist, which can aid in follow-on behavior such as brute-forcing, spear-phishing attacks, or account takeovers (e.g., [Valid Accounts](https://attack.mitre.org/techniques/T1078)). Adversaries may use several methods to enumerate accounts, including abuse of existing tools, built-in commands, and potential misconfigurations that leak account names and roles or permissions in the targeted environment. For examples, cloud environments typically provide easily accessible interfaces to obtain user lists.(Citation: AWS List Users)(Citation: Google Cloud - IAM Servie Accounts List API) On hosts, adversaries can use default [PowerShell](https://attack.mitre.org/techniques/T1059/001) and other command line functionality to identify accounts. Information about email addresses and accounts may also be extracted by searching an infected system's files.

Name

T1033

ID

T1033

Description

Adversaries may attempt to identify the primary user, currently logged in user, set of users that commonly uses a system, or whether a user is actively using the system. They may do this, for example, by retrieving account usernames or by using [OS Credential Dumping](https://attack.mitre.org/techniques/T1003). The information may be collected in a number of different ways using other Discovery techniques, because user and username details are prevalent throughout a system and include running process ownership, file/directory ownership, session information, and system logs. Adversaries may use the information from [System Owner/User Discovery](https://attack.mitre.org/techniques/T1033) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Various utilities and commands may acquire this information, including `whoami`. In macOS and Linux, the currently logged in user can be identified with `w` and `who`. On macOS the `dscl . list /Users | grep -v '_'` command can also be used to enumerate user accounts. Environment variables, such as `%USERNAME%` and `$USER`, may also be used to access this information. On network devices, [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) commands such as `show users` and `show ssh` can be used to display users currently logged into the device.(Citation: show_ssh_users_cmd_cisco)(Citation: US-CERT TA18-106A Network Infrastructure Devices 2018)

Name

T1071

ID

T1071

Description

Adversaries may communicate using OSI application layer protocols to avoid detection/network filtering by blending in with existing traffic. Commands to the remote system, and often the results of those commands, will be embedded within the protocol traffic between the client and server. Adversaries may utilize many different protocols, including those used for web browsing, transferring files, electronic mail, or DNS. For connections that occur internally within an enclave (such as those between a proxy or pivot node and other nodes), commonly used protocols are SMB, SSH, or RDP.(Citation: Mandiant APT29 Eye Spy Email Nov 22)

Name

T1083

ID

T1083

Description

Adversaries may enumerate files and directories or may search in specific locations of a host or network share for certain information within a file system. Adversaries may use the information from [File and Directory Discovery](https://attack.mitre.org/techniques/T1083) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Many command shell utilities can be used to obtain this information. Examples include `dir`, `tree`, `ls`, `find`, and `locate`.(Citation: Windows Commands JPCERT) Custom tools may also be used to gather file and directory information and interact with the [Native API](https://attack.mitre.org/techniques/T1106). Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather file and directory information (e.g. `dir`, `show flash`, and/or `nvram`).(Citation: US-CERT-TA18-106A) Some files and directories may require elevated or specific user permissions to access.

Indicator

Name
telegroms.icu
Pattern Type
stix
Pattern
[domain-name:value = 'telegroms.icu']
Name
https://telegroms.icu/assets/download/ready.apk
Pattern Type
stix
Pattern
[url:value = 'https://telegroms.icu/assets/download/ready.apk']
Name
http://154.213.65.28:7771

Pattern Type
stix
Pattern
[url:value = 'http://154.213.65.28:7771']
Name
154.213.65.28
Pattern Type
stix
Pattern
[ipv4-addr:value = '154.213.65.28']

Malware

Name
android

Name
SpyMax

indicates

Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name

uses

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name
Name

based-on

Name
Name
Name
Name

Domain-Name

Value
telegroms.icu

IPv4-Addr

Value
154.213.65.28

Url

Value
https://telegroms.icu/assets/download/ready.apk
http://154.213.65.28:7771

External References

- https://labs.k7computing.com/index.php/spymax-an-android-rat-targets-telegram-users/
- https://otx.alienvault.com/pulse/667ecd82548e727132558c15