Apr 10 2024

NETMANAGEIT

Intelligence Report

Targets human rights defenders in North Africa with new malware

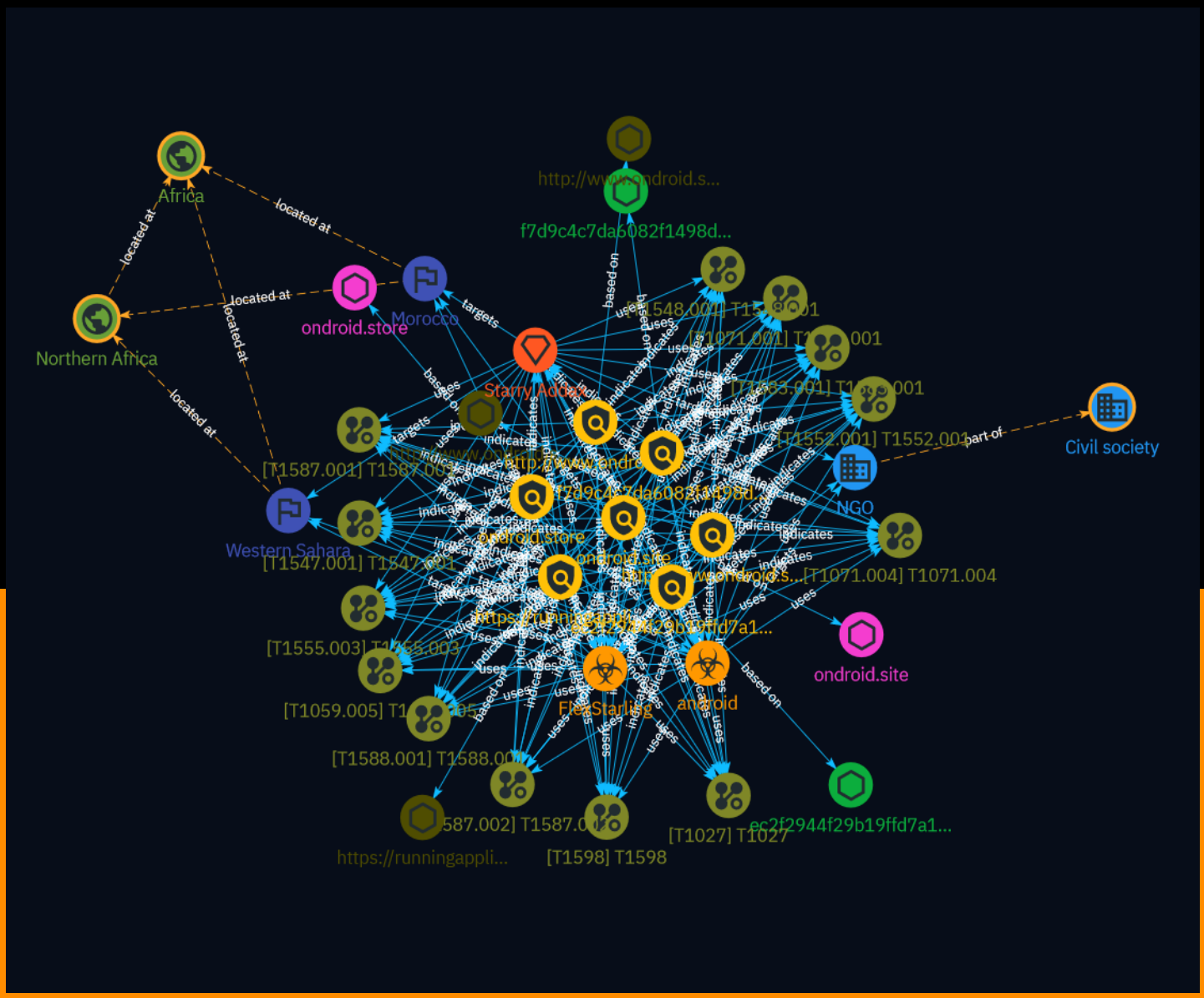

Table of contents

Overview

Entities

Observables

External References

Overview

Description

Cisco Talos revealed a novel threat actor dubbed 'Starry Addax' conducting phishing campaigns against individuals affiliated with the Sahrawi Arab Democratic Republic cause, particularly human rights activists in Morocco and Western Sahara. The group utilizes a custom Android malware called 'FlexStarling' masquerading as a legitimate news app to compromise devices and exfiltrate sensitive data. Additionally, Starry Addax employs credential harvesting tactics for Windows targets. This emerging campaign exhibits advanced capabilities and a focus on stealth operations.

Confidence

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

100 / 100

Content

N/A

Indicator

Name

https://runningapplications-b7dae-default-rtdb.firebaseio.com

Description

- **Unsafe:** False - **Server:** ESF - **Domain Rank:** 1840 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False - **Suspicious:** False - **Adult:** False - **Category:** Computers & Internet - **Domain Age:** {'human': '11 years ago', 'timestamp': 1350326781, 'iso': '2012-10-15T14:46:21-04:00'} - **IPQS: Domain:** runningapplications-b7dae-default-rtdb.firebaseio.com - **IPQS: IP Address:** 34.120.160.131

Pattern Type

stix

Pattern

[url:value = 'https://runningapplications-b7dae-default-rtdb.firebaseio.com']

Name

ondroid.store

Description

- **Unsafe:** True - **Server:** - **Domain Rank:** 0 - **DNS Valid:** False - **Parking:** False - **Spamming:** False - **Malware:** True - **Phishing:** True - **Suspicious:** True

- **Adult:** False - **Category:** N/A - **Domain Age:** {'human': '3 months ago', 'timestamp': 1704892557, 'iso': '2024-01-10T08:15:57-05:00'} - **IPQS: Domain:** ondroid.store - **IPQS: IP Address:** N/A

Pattern Type

stix

Pattern

[domain-name:value = 'ondroid.store']

Name

ondroid.site

Description

- **Unsafe:** False - **Server:** N/A - **Domain Rank:** 0 - **DNS Valid:** True - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False - **Suspicious:** True - **Adult:** False - **Category:** N/A - **Domain Age:** {'human': '2 months ago', 'timestamp': 1706801952, 'iso': '2024-02-01T10:39:12-05:00'} - **IPQS: Domain:** ondroid.site - **IPQS: IP Address:** 192.64.119.254

Pattern Type

stix

Pattern

[domain-name:value = 'ondroid.site']

Name

http://www.ondroid.store/ties5shizooQu1ei/

Description

- **Unsafe:** False - **Server:** - **Domain Rank:** 0 - **DNS Valid:** False - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False - **Suspicious:** True - **Adult:** False - **Category:** N/A - **Domain Age:** {'human': '3 months ago', 'timestamp': 1704892557, 'iso': '2024-01-10T08:15:57-05:00'} - **IPQS: Domain:** ondroid.store - **IPQS: IP Address:** N/A

Pattern Type

stix

Pattern

[url:value = 'http://www.ondroid.store/ties5shizooQu1ei/']

Name

http://www.ondroid.store/aL2mohh1

Description

- **Unsafe:** False - **Server:** N/A - **Domain Rank:** 0 - **DNS Valid:** False - **Parking:** False - **Spamming:** False - **Malware:** False - **Phishing:** False - **Suspicious:** True - **Adult:** False - **Category:** N/A - **Domain Age:** {'human': '3 months ago', 'timestamp': 1704892557, 'iso': '2024-01-10T08:15:57-05:00'} - **IPQS: Domain:** ondroid.store - **IPQS: IP Address:** N/A

Pattern Type

stix

Pattern

[url:value = 'http://www.ondroid.store/aL2mohh1']

Name
f7d9c4c7da6082f1498d41958b54d7aeffd0c674aab26db93309e88ca17c826c
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'f7d9c4c7da6082f1498d41958b54d7aeffd0c674aab26db93309e88ca17c826c']
Name
ec2f2944f29b19ffd7a1bb80ec3a98889ddf1c097130db6f30ad28c8bf9501b3
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = 'ec2f2944f29b19ffd7a1bb80ec3a98889ddf1c097130db6f30ad28c8bf9501b3']

Malware

Name
FlexStarling

Name
android

Intrusion-Set

Name
Starry Addax

Attack-Pattern

Name
T1059.005
ID
T1059.005
Description
Adversaries may abuse Visual Basic (VB) for execution. VB is a programming language created by Microsoft with interoperability with many Windows technologies such as [Component Object Model](https://attack.mitre.org/techniques/T1559/001) and the [Native API](https://attack.mitre.org/techniques/T1106) through the Windows API. Although tagged as legacy with no planned future evolutions, VB is integrated and supported in the .NET Framework and cross-platform .NET Core.(Citation: VB .NET Mar 2020)(Citation: VB Microsoft) Derivative languages based on VB have also been created, such as Visual Basic for Applications (VBA) and VBScript. VBA is an event-driven programming language built into Microsoft Office, as well as several third-party applications.(Citation: Microsoft VBA) (Citation: Wikipedia VBA) VBA enables documents to contain macros used to automate the execution of tasks and other functionality on the host. VBScript is a default scripting language on Windows hosts and can also be used in place of [JavaScript](https://attack.mitre.org/techniques/T1059/007) on HTML Application (HTA) webpages served to Internet Explorer (though most modern browsers do not come with VBScript support). (Citation: Microsoft VBScript) Adversaries may use VB payloads to execute malicious commands. Common malicious usage includes automating execution of behaviors with VBScript or embedding VBA content into [Spearphishing Attachment](https://attack.mitre.org/techniques/T1566/001) payloads (which may also involve [Mark-of-the-Web Bypass](https://attack.mitre.org/techniques/T1553/005) to enable execution).(Citation: Default VBS macros Blocking)

Name

T1071.001

ID

T1071.001

Description

Adversaries may communicate using application layer protocols associated with web traffic 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. Protocols such as HTTP/S(Citation: CrowdStrike Putter Panda) and WebSocket(Citation: Brazking-Websockets) that carry web traffic may be very common in environments. HTTP/S packets have many fields and headers in which data can be concealed. An adversary may abuse these protocols to communicate with systems under their control within a victim network while also mimicking normal, expected traffic.

Name

T1598

ID

T1598

Description

Adversaries may send phishing messages to elicit sensitive information that can be used during targeting. Phishing for information is an attempt to trick targets into divulging information, frequently credentials or other actionable information. Phishing for information is different from [Phishing](https://attack.mitre.org/techniques/T1566) in that the objective is gathering data from the victim rather than executing malicious code. All forms of phishing are electronically delivered social engineering. Phishing can be targeted, known as spearphishing. In spearphishing, a specific individual, company, or industry will be targeted by the adversary. More generally, adversaries can conduct non-targeted

phishing, such as in mass credential harvesting campaigns. Adversaries may also try to obtain information directly through the exchange of emails, instant messages, or other electronic conversation means.(Citation: ThreatPost Social Media Phishing)(Citation: TrendMictro Phishing)(Citation: PCMag FakeLogin)(Citation: Sophos Attachment)(Citation: GitHub Phishery) Victims may also receive phishing messages that direct them to call a phone number where the adversary attempts to collect confidential information.(Citation: Avertium callback phishing) Phishing for information frequently involves social engineering techniques, such as posing as a source with a reason to collect information (ex: [Establish Accounts](https://attack.mitre.org/techniques/T1585) or [Compromise Accounts](https://attack.mitre.org/techniques/T1586)) and/or sending multiple, seemingly urgent messages. Another way to accomplish this is by forging or spoofing(Citation: Proofpoint-spoof) the identity of the sender which can be used to fool both the human recipient as well as automated security tools.(Citation: cyberproof-double-bounce) Phishing for information may also involve evasive techniques, such as removing or manipulating emails or metadata/headers from compromised accounts being abused to send messages (e.g., [Email Hiding Rules](https://attack.mitre.org/techniques/T1564/008)). (Citation: Microsoft OAuth Spam 2022)(Citation: Palo Alto Unit 42 VBA Infostealer 2014)

Name

T1547.001

ID

T1547.001

Description

Adversaries may achieve persistence by adding a program to a startup folder or referencing it with a Registry run key. Adding an entry to the "run keys" in the Registry or startup folder will cause the program referenced to be executed when a user logs in. (Citation: Microsoft Run Key) These programs will be executed under the context of the user and will have the account's associated permissions level. The following run keys are created by default on Windows systems: * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunOnce` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce` Run keys may exist under multiple hives.(Citation: Microsoft Wow6432Node 2018)(Citation: Malwarebytes Wow6432Node 2016) The `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnceEx` is also

available but is not created by default on Windows Vista and newer. Registry run key entries can reference programs directly or list them as a dependency.(Citation: Microsoft Run Key) For example, it is possible to load a DLL at logon using a "Depend" key with RunOnceEx: `reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\RunOnceEx\0001\Depend /v 1 /d "C:\temp\evil[.]dll"` (Citation: Oddvar Moe RunOnceEx Mar 2018) Placing a program within a startup folder will also cause that program to execute when a user logs in. There is a startup folder location for individual user accounts as well as a system-wide startup folder that will be checked regardless of which user account logs in. The startup folder path for the current user is `C:\Users\\[Username]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup`. The startup folder path for all users is `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp`. The following Registry keys can be used to set startup folder items for persistence: * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders` * `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders` * `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders` The following Registry keys can control automatic startup of services during boot: * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServicesOnce` * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\RunServices` Using policy settings to specify startup programs creates corresponding values in either of two Registry keys: * `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run` * `HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Policies\Explorer\Run` Programs listed in the load value of the registry key `HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows` run automatically for the currently logged-on user. By default, the multistring `BootExecute` value of the registry key `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\Session Manager` is set to `autocheck autochk *`. This value causes Windows, at startup, to check the file-system integrity of the hard disks if the system has been shut down abnormally. Adversaries can add other programs or processes to this registry value which will automatically launch at boot. Adversaries can use these configuration locations to execute malware, such as remote access tools, to maintain persistence through system reboots. Adversaries may also use [Masquerading](https://attack.mitre.org/techniques/T1036) to make the Registry entries look as if they are associated with legitimate programs.

Name

T1552.001

ID

T1552.001

Description

Adversaries may search local file systems and remote file shares for files containing insecurely stored credentials. These can be files created by users to store their own credentials, shared credential stores for a group of individuals, configuration files containing passwords for a system or service, or source code/binary files containing embedded passwords. It is possible to extract passwords from backups or saved virtual machines through [OS Credential Dumping](https://attack.mitre.org/techniques/T1003). (Citation: CG 2014) Passwords may also be obtained from Group Policy Preferences stored on the Windows Domain Controller. (Citation: SRD GPP) In cloud and/or containerized environments, authenticated user and service account credentials are often stored in local configuration and credential files.(Citation: Unit 42 Hildegard Malware) They may also be found as parameters to deployment commands in container logs.(Citation: Unit 42 Unsecured Docker Daemons) In some cases, these files can be copied and reused on another machine or the contents can be read and then used to authenticate without needing to copy any files.(Citation: Specter Ops - Cloud Credential Storage)

Name

T1588.001

ID

T1588.001

Description

Adversaries may buy, steal, or download malware that can be used during targeting. Malicious software can include payloads, droppers, post-compromise tools, backdoors, packers, and C2 protocols. Adversaries may acquire malware to support their operations,

obtaining a means for maintaining control of remote machines, evading defenses, and executing post-compromise behaviors. In addition to downloading free malware from the internet, adversaries may purchase these capabilities from third-party entities. Third-party entities can include technology companies that specialize in malware development, criminal marketplaces (including Malware-as-a-Service, or MaaS), or from individuals. In addition to purchasing malware, adversaries may steal and repurpose malware from third-party entities (including other adversaries).

Name

T1548.001

ID

T1548.001

Description

An adversary may abuse configurations where an application has the setuid or setgid bits set in order to get code running in a different (and possibly more privileged) user's context. On Linux or macOS, when the setuid or setgid bits are set for an application binary, the application will run with the privileges of the owning user or group respectively.(Citation: setuid man page) Normally an application is run in the current user's context, regardless of which user or group owns the application. However, there are instances where programs need to be executed in an elevated context to function properly, but the user running them may not have the specific required privileges. Instead of creating an entry in the sudoers file, which must be done by root, any user can specify the setuid or setgid flag to be set for their own applications (i.e. [Linux and Mac File and Directory Permissions Modification](https://attack.mitre.org/techniques/T1222/002)). The `chmod` command can set these bits with bitmasking, `chmod 4777 [file]` or via shorthand naming, `chmod u+s [file]`. This will enable the setuid bit. To enable the setgid bit, `chmod 2775` and `chmod g+s` can be used. Adversaries can use this mechanism on their own malware to make sure they're able to execute in elevated contexts in the future.(Citation: OSX Keydnap malware) This abuse is often part of a "shell escape" or other actions to bypass an execution environment with restricted permissions. Alternatively, adversaries may choose to find and target vulnerable binaries with the setuid or setgid bits already enabled (i.e. [File and Directory Discovery](https://attack.mitre.org/techniques/T1083)). The setuid and setguid bits are indicated with an "s" instead of an "x" when viewing a file's attributes via `ls -l`. The `find` command can also be used to search for such files. For example, `find / -perm +4000 2>/dev/null` can be used to find files with setuid set and

`find / -perm +2000 2>/dev/null` may be used for setgid. Binaries that have these bits set may then be abused by adversaries.(Citation: GTFOBins Suid)

Name

T1027

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](https://attack.mitre.org/techniques/T1140) for [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](https://attack.mitre.org/techniques/T1027/010) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

T1555.003

ID

T1555.003

Description

Adversaries may acquire credentials from web browsers by reading files specific to the target browser.(Citation: Talos Olympic Destroyer 2018) Web browsers commonly save credentials such as website usernames and passwords so that they do not need to be entered manually in the future. Web browsers typically store the credentials in an encrypted format within a credential store; however, methods exist to extract plaintext credentials from web browsers. For example, on Windows systems, encrypted credentials may be obtained from Google Chrome by reading a database file, `AppData\Local\Google\Chrome\User Data\Default\Login Data` and executing a SQL query: `SELECT action_url, username_value, password_value FROM logins;`. The plaintext password can then be obtained by passing the encrypted credentials to the Windows API function `CryptUnprotectData`, which uses the victim's cached logon credentials as the decryption key.(Citation: Microsoft CryptUnprotectData April 2018) Adversaries have executed similar procedures for common web browsers such as FireFox, Safari, Edge, etc. (Citation: Proofpoint Vega Credential Stealer May 2018)(Citation: FireEye HawkEye Malware July 2017) Windows stores Internet Explorer and Microsoft Edge credentials in Credential Lockers managed by the [Windows Credential Manager](https://attack.mitre.org/techniques/T1555/004). Adversaries may also acquire credentials by searching web browser process memory for patterns that commonly match credentials.(Citation: GitHub Mimikittenz July 2016) After acquiring credentials from web browsers, adversaries may attempt to recycle the credentials across different systems and/or accounts in order to expand access. This can result in significantly furthering an adversary's objective in cases where credentials gained from web browsers overlap with privileged accounts (e.g. domain administrator).

Name

T1583.001

ID

T1583.001

Description

Adversaries may acquire domains that can be used during targeting. Domain names are the human readable names used to represent one or more IP addresses. They can be purchased or, in some cases, acquired for free. Adversaries may use acquired domains for a variety of purposes, including for [Phishing](https://attack.mitre.org/techniques/T1566), [Drive-by Compromise](https://attack.mitre.org/techniques/T1189), and Command and Control.(Citation: CISA MSS Sep 2020) Adversaries may choose domains that are similar to legitimate domains, including through use of homoglyphs or use of a different top-level domain (TLD).(Citation: FireEye APT28)(Citation: PaypalScam) Typosquatting may be used to aid in delivery of payloads via [Drive-by Compromise](https://attack.mitre.org/techniques/T1189). Adversaries may also use internationalized domain names (IDNs) and different character sets (e.g. Cyrillic, Greek, etc.) to execute "IDN homograph attacks," creating visually similar lookalike domains used to deliver malware to victim machines.(Citation: CISA IDN ST05-016)(Citation: tt_httrack_fake_domains)(Citation: tt_obliqueRAT)(Citation: httrack_unhcr)(Citation: lazgroup_idn_phishing) Adversaries may also acquire and repurpose expired domains, which may be potentially already allowlisted/trusted by defenders based on an existing reputation/history.(Citation: Categorisation_not_boundary)(Citation: Domain_Steal_CC)(Citation: Redirectors_Domain_Fronting)(Citation: bypass_webproxy_filtering) Domain registrars each maintain a publicly viewable database that displays contact information for every registered domain. Private WHOIS services display alternative information, such as their own company data, rather than the owner of the domain. Adversaries may use such private WHOIS services to obscure information about who owns a purchased domain. Adversaries may further interrupt efforts to track their infrastructure by using varied registration information and purchasing domains with different domain registrars.(Citation: Mandiant APT1)

Name

T1587.002

ID

T1587.002

Description

Adversaries may create self-signed code signing certificates that can be used during targeting. Code signing is the process of digitally signing executables and scripts to confirm the software author and guarantee that the code has not been altered or corrupted. Code signing provides a level of authenticity for a program from the developer and a guarantee that the program has not been tampered with.(Citation: Wikipedia Code

Signing) Users and/or security tools may trust a signed piece of code more than an unsigned piece of code even if they don't know who issued the certificate or who the author is. Prior to [Code Signing](https://attack.mitre.org/techniques/T1553/002), adversaries may develop self-signed code signing certificates for use in operations.

Name

T1587.001

ID

T1587.001

Description

Adversaries may develop malware and malware components that can be used during targeting. Building malicious software can include the development of payloads, droppers, post-compromise tools, backdoors (including backdoored images), packers, C2 protocols, and the creation of infected removable media. Adversaries may develop malware to support their operations, creating a means for maintaining control of remote machines, evading defenses, and executing post-compromise behaviors.(Citation: Mandiant APT1) (Citation: Kaspersky Sofacy)(Citation: ActiveMalwareEnergy)(Citation: FBI Flash FIN7 USB) As with legitimate development efforts, different skill sets may be required for developing malware. The skills needed may be located in-house, or may need to be contracted out. Use of a contractor may be considered an extension of that adversary's malware development capabilities, provided the adversary plays a role in shaping requirements and maintains a degree of exclusivity to the malware. Some aspects of malware development, such as C2 protocol development, may require adversaries to obtain additional infrastructure. For example, malware developed that will communicate with Twitter for C2, may require use of [Web Services](https://attack.mitre.org/techniques/T1583/006).(Citation: FireEye APT29)

Name

T1071.004

ID

T1071.004

Description

Adversaries may communicate using the Domain Name System (DNS) application layer protocol 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. The DNS protocol serves an administrative function in computer networking and thus may be very common in environments. DNS traffic may also be allowed even before network authentication is completed. DNS packets contain many fields and headers in which data can be concealed. Often known as DNS tunneling, adversaries may abuse DNS to communicate with systems under their control within a victim network while also mimicking normal, expected traffic. (Citation: PAN DNS Tunneling)(Citation: Medium DnsTunneling)

Country

Name
Western Sahara

Name
Morocco

Region

Name
Northern Africa

Name
Africa

Sector

Name
NGO
Description
A legally constituted non-commercial organization created by natural or legal persons with no participation or representation of any government.
Name
Civil society
Description
The general public and all non-governmental entities, or individuals independent of governments, which may be linked by interests or activities aiming at promoting the interests and the will of citizens.

Url

Value
https://runningapplications-b7dae-default-rtdb.firebaseio.com
http://www.ondroid.store/ties5shizooQu1ei/
http://www.ondroid.store/aL2mohh1

Domain-Name

Value
ondroid.store
ondroid.site

StixFile

Value
f7d9c4c7da6082f1498d41958b54d7aeffd0c674aab26db93309e88ca17c826c
ec2f2944f29b19ffd7a1bb80ec3a98889ddf1c097130db6f30ad28c8bf9501b3

External References

- https://blog.talosintelligence.com/starry-addax/
- https://otx.alienvault.com/pulse/6616902ec7eb8d65da794c46