Sep 25 2023

NETMANAGEIT

Intelligence Report

Backchannel Diplomacy: APT29’s Rapidly Evolving Diplomatic Phishing Operations

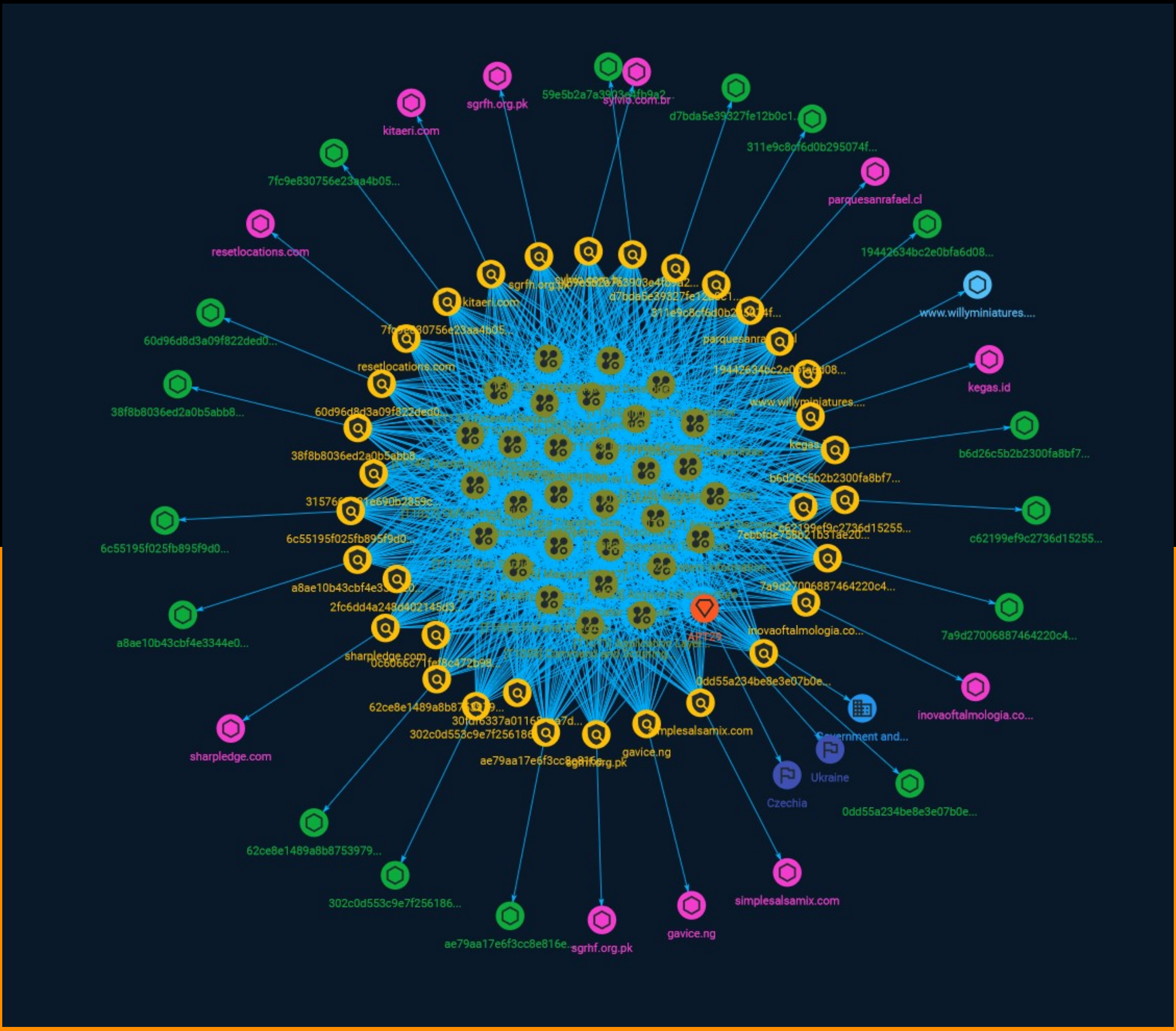

Table of contents

Overview

Entities

Observables

External References

Overview

Description

APT29’s pace of operations and emphasis on Ukraine increased in the first half of 2023 as Kyiv launched its counteroffensive, pointing to the SVR’s central role in collecting intelligence concerning the current pivotal phase of the war.

Confidence

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

15 / 100

Attack-Pattern

Name

Process Discovery

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. 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

Data Transfer Size Limits

ID

T1030

Description

An adversary may exfiltrate data in fixed size chunks instead of whole files or limit packet sizes below certain thresholds. This approach may be used to avoid triggering network data transfer threshold alerts.

Name

Boot or Logon Autostart Execution

ID

T1547

Description

Adversaries may configure system settings to automatically execute a program during system boot or logon to maintain persistence or gain higher-level privileges on compromised systems. Operating systems may have mechanisms for automatically running a program on system boot or account logon.(Citation: Microsoft Run Key)(Citation: MSDN Authentication Packages)(Citation: Microsoft TimeProvider)(Citation: Cylance Reg Persistence Sept 2013)(Citation: Linux Kernel Programming) These mechanisms may include automatically executing programs that are placed in specially designated directories or are referenced by repositories that store configuration information, such as the Windows Registry. An adversary may achieve the same goal by modifying or extending features of the kernel. Since some boot or logon autostart programs run with higher privileges, an adversary may leverage these to elevate privileges.

Name

Virtualization/Sandbox Evasion

ID

T1497

Description

Adversaries may employ various means to detect and avoid virtualization and analysis environments. This may include changing behaviors based on the results of checks for the presence of artifacts indicative of a virtual machine environment (VME) or sandbox. If the adversary detects a VME, they may alter their malware to disengage from the victim or conceal the core functions of the implant. They may also search for VME artifacts before dropping secondary or additional payloads. Adversaries may use the information learned from [Virtualization/Sandbox Evasion](https://attack.mitre.org/techniques/T1497) during automated discovery to shape follow-on behaviors.(Citation: Deloitte Environment Awareness) Adversaries may use several methods to accomplish [Virtualization/Sandbox Evasion](https://attack.mitre.org/techniques/T1497) such as checking for security monitoring tools (e.g., Sysinternals, Wireshark, etc.) or other system artifacts associated with analysis or virtualization. Adversaries may also check for legitimate user activity to help determine if it is in an analysis environment. Additional methods include use of sleep timers or loops within malware code to avoid operating within a temporary sandbox.(Citation: Unit 42 Pirpi July 2015)

Name

Compromise Infrastructure

ID

T1584

Description

Adversaries may compromise third-party infrastructure that can be used during targeting. Infrastructure solutions include physical or cloud servers, domains, and third-party web and DNS services. Instead of buying, leasing, or renting infrastructure an adversary may compromise infrastructure and use it during other phases of the adversary lifecycle.(Citation: Mandiant APT1)(Citation: ICANNDomainNameHijacking)(Citation: Talos DNSpionage Nov 2018)(Citation: FireEye EPS Awakens Part 2) Additionally, adversaries may compromise numerous machines to form a botnet they can leverage. Use of compromised infrastructure allows adversaries to stage, launch, and execute operations. Compromised infrastructure can help adversary operations blend in with traffic that is seen as normal, such as contact with high reputation or trusted sites. For example, adversaries may leverage compromised infrastructure (potentially also in conjunction with [Digital Certificates](https://attack.mitre.org/techniques/T1588/004)) to further blend in and support staged information gathering and/or [Phishing](https://attack.mitre.org/techniques/T1566) campaigns.(Citation: FireEye DNS Hijack 2019) Additionally, adversaries may also compromise infrastructure to support [Proxy](https://attack.mitre.org/techniques/T1090).(Citation: amnesty_nso_pegasus) By using compromised infrastructure, adversaries may make it difficult to tie their actions back to them. Prior to targeting, adversaries may compromise the infrastructure of other adversaries.(Citation: NSA NCSC Turla OilRig)

Name

Query Registry

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

Masquerading

ID

T1036

Description

Adversaries may attempt to manipulate features of their artifacts to make them appear legitimate or benign to users and/or security tools. Masquerading occurs when the name or location of an object, legitimate or malicious, is manipulated or abused for the sake of evading defenses and observation. This may include manipulating file metadata, tricking users into misidentifying the file type, and giving legitimate task or service names. Renaming abusable system utilities to evade security monitoring is also a form of [Masquerading](https://attack.mitre.org/techniques/T1036).(Citation: LOLBAS Main Site)

Name

Process Injection

ID

T1055

Description

Adversaries may inject code into processes in order to evade process-based defenses as well as possibly elevate privileges. Process injection is a method of executing arbitrary code in the address space of a separate live process. Running code in the context of another process may allow access to the process's memory, system/network resources, and possibly elevated privileges. Execution via process injection may also evade detection from security products since the execution is masked under a legitimate process. There are many different ways to inject code into a process, many of which abuse legitimate functionalities. These implementations exist for every major OS but are typically platform specific. More sophisticated samples may perform multiple process injections to segment modules and further evade detection, utilizing named pipes or other inter-process communication (IPC) mechanisms as a communication channel.

Name

Scheduled Task/Job

ID

T1053

Description

Adversaries may abuse task scheduling functionality to facilitate initial or recurring execution of malicious code. Utilities exist within all major operating systems to schedule programs or scripts to be executed at a specified date and time. A task can also be scheduled on a remote system, provided the proper authentication is met (ex: RPC and file and printer sharing in Windows environments). Scheduling a task on a remote system typically may require being a member of an admin or otherwise privileged group on the remote system.(Citation: TechNet Task Scheduler Security) Adversaries may use task scheduling to execute programs at system startup or on a scheduled basis for persistence. These mechanisms can also be abused to run a process under the context of a specified account (such as one with elevated permissions/privileges). Similar to [System Binary Proxy Execution](https://attack.mitre.org/techniques/T1218), adversaries have also abused task scheduling to potentially mask one-time execution under a trusted system process. (Citation: ProofPoint Serpent)

Name

Non-Standard Port

ID

T1571

Description

Adversaries may communicate using a protocol and port pairing that are typically not associated. For example, HTTPS over port 8088(Citation: Symantec Elfin Mar 2019) or port 587(Citation: Fortinet Agent Tesla April 2018) as opposed to the traditional port 443. Adversaries may make changes to the standard port used by a protocol to bypass filtering or muddle analysis/parsing of network data. Adversaries may also make changes to victim systems to abuse non-standard ports. For example, Registry keys and other configuration settings can be used to modify protocol and port pairings.(Citation: change_rdp_port_conti)

Name

Encrypted Channel

ID

T1573

Description

Adversaries may employ a known encryption algorithm to conceal command and control traffic rather than relying on any inherent protections provided by a communication protocol. Despite the use of a secure algorithm, these implementations may be vulnerable to reverse engineering if secret keys are encoded and/or generated within malware samples/configuration files.

Name

Indicator Removal

ID

T1070

Description

Adversaries may delete or modify artifacts generated within systems to remove evidence of their presence or hinder defenses. Various artifacts may be created by an adversary or something that can be attributed to an adversary’s actions. Typically these artifacts are used as defensive indicators related to monitored events, such as strings from downloaded files, logs that are generated from user actions, and other data analyzed by defenders. Location, format, and type of artifact (such as command or login history) are often specific to each platform. Removal of these indicators may interfere with event collection, reporting, or other processes used to detect intrusion activity. This may compromise the integrity of security solutions by causing notable events to go unreported. This activity may also impede forensic analysis and incident response, due to lack of sufficient data to determine what occurred.

Name

Phishing

ID

T1566

Description

Adversaries may send phishing messages to gain access to victim systems. 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 malware spam campaigns. Adversaries may send victims emails containing malicious attachments or links, typically to execute malicious code on victim systems. Phishing may also be conducted via third-party services, like social media platforms. Phishing may also involve social engineering techniques, such as posing as a trusted source, as well as 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) 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) Victims may also receive phishing messages that instruct them to call a phone number where they are directed to visit a malicious URL, download malware,(Citation: sygnia Luna Month)(Citation: CISA Remote Monitoring and Management Software) or install adversary-accessible remote management tools onto their computer (i.e., [User Execution](https://attack.mitre.org/techniques/T1204)).(Citation: Unit42 Luna Moth)

Name

Software Discovery

ID

T1518

Description

Adversaries may attempt to get a listing of software and software versions that are installed on a system or in a cloud environment. Adversaries may use the information from [Software Discovery](https://attack.mitre.org/techniques/T1518) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Adversaries may attempt to enumerate software for a variety of reasons, such as figuring out what security measures are present or if the compromised system has a version of software that is vulnerable to [Exploitation for Privilege Escalation](https://attack.mitre.org/techniques/T1068).

Name

Modify Registry

ID

T1112

Description

Adversaries may interact with the Windows Registry to hide configuration information within Registry keys, remove information as part of cleaning up, or as part of other techniques to aid in persistence and execution. Access to specific areas of the Registry depends on account permissions, some requiring administrator-level access. The built-in Windows command-line utility [Reg](https://attack.mitre.org/software/S0075) may be used for local or remote Registry modification. (Citation: Microsoft Reg) Other tools may also be used, such as a remote access tool, which may contain functionality to interact with the Registry through the Windows API. Registry modifications may also include actions to hide keys, such as prepending key names with a null character, which will cause an error and/or be ignored when read via [Reg](https://attack.mitre.org/software/S0075) or other utilities using the Win32 API. (Citation: Microsoft Reghide NOV 2006) Adversaries may abuse these pseudo-hidden keys to conceal payloads/commands used to maintain persistence. (Citation: TrendMicro POWELIKS AUG 2014) (Citation: SpectorOps Hiding Reg Jul 2017) The Registry of a remote system may be modified to aid in execution of files as part of lateral movement. It requires the remote Registry service to be running on the target system. (Citation: Microsoft Remote) Often [Valid Accounts](https://attack.mitre.org/techniques/T1078) are required, along with access to the remote system's [SMB/Windows Admin Shares](https://attack.mitre.org/techniques/T1021/002) for RPC communication.

Name

User Execution

ID

T1204

Description

An adversary may rely upon specific actions by a user in order to gain execution. Users may be subjected to social engineering to get them to execute malicious code by, for example, opening a malicious document file or link. These user actions will typically be observed as follow-on behavior from forms of [Phishing](https://attack.mitre.org/techniques/T1566). While [User Execution](https://attack.mitre.org/techniques/T1204) frequently occurs shortly after Initial Access it may occur at other phases of an intrusion, such as when an adversary places a file in a shared directory or on a user's desktop hoping that a user will click on it. This activity may also be seen shortly after [Internal Spearphishing](https://attack.mitre.org/techniques/T1534). Adversaries may also deceive users into performing actions such as enabling [Remote Access Software](https://attack.mitre.org/techniques/T1219), allowing direct control of the system to the adversary, or downloading and executing malware for [User Execution](https://attack.mitre.org/techniques/T1204). For example, tech support scams can be facilitated through [Phishing](https://attack.mitre.org/techniques/T1566), vishing, or various forms of user interaction. Adversaries can use a combination of these methods, such as spoofing and promoting toll-free numbers or call centers that are used to direct victims to malicious websites, to deliver and execute payloads containing malware or [Remote Access Software](https://attack.mitre.org/techniques/T1219).(Citation: Telephone Attack Delivery)

Name

Acquire Infrastructure

ID

T1583

Description

Adversaries may buy, lease, or rent infrastructure that can be used during targeting. A wide variety of infrastructure exists for hosting and orchestrating adversary operations. Infrastructure solutions include physical or cloud servers, domains, and third-party web services.(Citation: TrendmicroHideoutsLease) Additionally, botnets are available for rent or purchase. Use of these infrastructure solutions allows adversaries to stage, launch, and execute operations. Solutions may help adversary operations blend in with traffic that is seen as normal, such as contacting third-party web services or acquiring infrastructure to support [Proxy](https://attack.mitre.org/techniques/T1090).(Citation: amnesty_nso_pegasus) Depending on the implementation, adversaries may use infrastructure that makes it difficult to physically tie back to them as well as utilize infrastructure that can be rapidly provisioned, modified, and shut down.

Name

Obfuscated Files or Information

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

Ingress Tool Transfer

ID

T1105

Description

Adversaries may transfer tools or other files from an external system into a compromised environment. Tools or files may be copied from an external adversary-controlled system to the victim network through the command and control channel or through alternate protocols such as [ftp](https://attack.mitre.org/software/S0095). Once present, adversaries may also transfer/spread tools between victim devices within a compromised environment (i.e. [Lateral Tool Transfer](https://attack.mitre.org/techniques/T1570)). Files can also be transferred using various [Web Service](https://attack.mitre.org/techniques/T1102)s as well as native or otherwise present tools on the victim system.(Citation: PTSecurity Cobalt Dec 2016) On Windows, adversaries may use various utilities to download tools, such as `copy`, `finger`, [certutil](https://attack.mitre.org/software/S0160), and [PowerShell](https://attack.mitre.org/techniques/T1059/001) commands such as `IEX(New-Object Net.WebClient).downloadString()` and `Invoke-WebRequest`. On Linux and macOS systems, a variety of utilities also exist, such as `curl`, `scp`, `sftp`, `tftp`, `rsync`, `finger`, and `wget`.(Citation: t1105_lolbas)

Name

Command and Scripting Interpreter

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

External Remote Services

ID

T1133

Description

Adversaries may leverage external-facing remote services to initially access and/or persist within a network. Remote services such as VPNs, Citrix, and other access mechanisms allow users to connect to internal enterprise network resources from external locations. There are often remote service gateways that manage connections and credential authentication for these services. Services such as [Windows Remote Management](https://attack.mitre.org/techniques/T1021/006) and [VNC](https://attack.mitre.org/techniques/T1021/005) can also be used externally.(Citation: MacOS VNC software for Remote Desktop) Access to [Valid Accounts](https://attack.mitre.org/techniques/T1078) to use the service is often a requirement, which could be obtained through credential pharming or by obtaining the credentials from users after compromising the enterprise network.(Citation: Volexity Virtual Private Keylogging) Access to remote services may be used as a redundant or persistent access mechanism during an operation. Access may also be gained through an exposed service that doesn't require authentication. In containerized environments, this may include an exposed Docker API, Kubernetes API server, kubelet, or web application such as the Kubernetes dashboard.(Citation: Trend Micro Exposed Docker Server)(Citation: Unit 42 Hildegard Malware)

Name

Account Discovery

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. 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

Web Service

ID

T1102

Description

Adversaries may use an existing, legitimate external Web service as a means for relaying data to/from a compromised system. Popular websites and social media acting as a mechanism for C2 may give a significant amount of cover due to the likelihood that hosts within a network are already communicating with them prior to a compromise. Using common services, such as those offered by Google or Twitter, makes it easier for adversaries to hide in expected noise. Web service providers commonly use SSL/TLS encryption, giving adversaries an added level of protection. Use of Web services may also protect back-end C2 infrastructure from discovery through malware binary analysis while also enabling operational resiliency (since this infrastructure may be dynamically changed).

Name

Trusted Developer Utilities Proxy Execution

ID

T1127

Description

Adversaries may take advantage of trusted developer utilities to proxy execution of malicious payloads. There are many utilities used for software development related tasks that can be used to execute code in various forms to assist in development, debugging, and reverse engineering.(Citation: engima0x3 DNX Bypass)(Citation: engima0x3 RCSI Bypass)(Citation: Exploit Monday WinDbg)(Citation: LOLBAS Tracker) These utilities may often be signed with legitimate certificates that allow them to execute on a system and proxy execution of malicious code through a trusted process that effectively bypasses application control solutions.

Name

Application Layer Protocol

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.

Name

Deobfuscate/Decode Files or Information

ID

T1140

Description

Adversaries may use [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system. One such example is the use of [certutil](https://attack.mitre.org/software/S0160) to decode a remote access tool portable executable file that has been hidden inside a certificate file.(Citation: Malwarebytes Targeted Attack against Saudi Arabia) Another example is using the Windows `copy /b` command to reassemble binary fragments into a malicious payload.(Citation: Carbon Black Obfuscation Sept 2016) Sometimes a user's action may be required to open it for deobfuscation or decryption as part of [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)

Name

Obtain Capabilities

ID

T1588

Description

Adversaries may buy and/or steal capabilities that can be used during targeting. Rather than developing their own capabilities in-house, adversaries may purchase, freely download, or steal them. Activities may include the acquisition of malware, software (including licenses), exploits, certificates, and information relating to vulnerabilities. Adversaries may obtain capabilities to support their operations throughout numerous phases of the adversary lifecycle. In addition to downloading free malware, software, and exploits from the internet, adversaries may purchase these capabilities from third-party entities. Third-party entities can include technology companies that specialize in malware and exploits, criminal marketplaces, or from individuals.(Citation: NationsBuying)(Citation: PegasusCitizenLab) In addition to purchasing capabilities, adversaries may steal capabilities from third-party entities (including other adversaries). This can include stealing software licenses, malware, SSL/TLS and code-signing certificates, or raiding closed databases of vulnerabilities or exploits.(Citation: DiginotarCompromise)

Name

File and Directory Discovery

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)

Name

Stage Capabilities

ID

T1608

Description

Adversaries may upload, install, or otherwise set up capabilities that can be used during targeting. To support their operations, an adversary may need to take capabilities they developed ([Develop Capabilities](https://attack.mitre.org/techniques/T1587)) or obtained ([Obtain Capabilities](https://attack.mitre.org/techniques/T1588)) and stage them on infrastructure under their control. These capabilities may be staged on infrastructure that was previously purchased/rented by the adversary ([Acquire Infrastructure](https://attack.mitre.org/techniques/T1583)) or was otherwise compromised by them ([Compromise Infrastructure](https://attack.mitre.org/techniques/T1584)). Capabilities may also be staged on web services, such as GitHub or Pastebin, or on Platform-as-a-Service (PaaS) offerings that enable users to easily provision applications.(Citation: Volexity Ocean Lotus November 2020)(Citation: Dragos Heroku Watering Hole)(Citation: Malwarebytes Heroku Skimmers)(Citation: Netskope GCP Redirection)(Citation: Netskope Cloud Phishing) Staging of capabilities can aid the adversary in a number of initial access and post-compromise behaviors, including (but not limited to): * Staging web resources necessary to conduct [Drive-by Compromise](https://attack.mitre.org/techniques/T1189) when a user browses to a site.(Citation: FireEye CFR Watering Hole 2012)(Citation: Gallagher 2015)(Citation: ATT ScanBox) * Staging web resources for a link target to be used with spearphishing.(Citation: Malwarebytes Silent Librarian October 2020)(Citation: Proofpoint TA407 September 2019) * Uploading malware or tools to a location accessible to a victim network to enable [Ingress Tool Transfer](https://attack.mitre.org/techniques/T1105).(Citation: Volexity Ocean Lotus November 2020) * Installing a previously acquired SSL/TLS certificate to use to encrypt command and control traffic (ex: [Asymmetric Cryptography](https://attack.mitre.org/techniques/T1573/002) with [Web Protocols](https://attack.mitre.org/techniques/T1071/001)).(Citation: DigiCert Install SSL Cert)

Name

System Information Discovery

ID

T1082

Description

An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from [System Information Discovery](https://attack.mitre.org/techniques/T1082) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Tools such as [Systeminfo](https://attack.mitre.org/software/S0096) can be used to gather detailed system information. If running with privileged access, a breakdown of system data can be gathered through the `systemsetup` configuration tool on macOS. As an example, adversaries with user-level access can execute the `df -aH` command to obtain currently mounted disks and associated freely available space. Adversaries may also leverage a [Network Device CLI](https://attack.mitre.org/techniques/T1059/008) on network devices to gather detailed system information (e.g. `show version`).(Citation: US-CERT-TA18-106A) [System Information Discovery](https://attack.mitre.org/techniques/T1082) combined with information gathered from other forms of discovery and reconnaissance can drive payload development and concealment.(Citation: OSX.FairyTale)(Citation: 20 macOS Common Tools and Techniques) Infrastructure as a Service (IaaS) cloud providers such as AWS, GCP, and Azure allow access to instance and virtual machine information via APIs. Successful authenticated API calls can return data such as the operating system platform and status of a particular instance or the model view of a virtual machine.(Citation: Amazon Describe Instance)(Citation: Google Instances Resource)(Citation: Microsoft Virutal Machine API)

Sector

Name
Government and administrations
Description
Civilian government institutions and administrations of the executive and legislative branches. The diplomatic and judicial branches are not included.

Indicator

Name
38f8b8036ed2a0b5abb8fbf264ee6fd2b82dcd917f60d9f1d8f18d07c26b1534
Description
SHA256 of 53270b3968004cb48dac1a1b239ed23d
Pattern Type
stix
Pattern
[file:hashes.'SHA-256' = '38f8b8036ed2a0b5abb8fbf264ee6fd2b82dcd917f60d9f1d8f18d07c26b1534']
Name
kegas.id
Pattern Type
stix
Pattern

[domain-name:value = 'kegas.id']

Name

302c0d553c9e7f2561864d79022b780a53ec0a5927e8962d883b88dde249d044

Description

invalid_XObject_js SHA256 of fc53c75289309ffb7f65a3513e7519eb

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '302c0d553c9e7f2561864d79022b780a53ec0a5927e8962d883b88dde249d044']

Name

sgrfh.org.pk

Pattern Type

stix

Pattern

[domain-name:value = 'sgrfh.org.pk']

Name

inovaoftalmologia.com.br

Pattern Type

stix

Pattern

[domain-name:value = 'inovaoftalmologia.com.br']

Name

a8ae10b43cbf4e3344e0184b33a699b19a29866bc1e41201ace1a995e8ca3149

Description

SHA256 of 9e51506816ad620c9e6474c52a9004a6

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'a8ae10b43cbf4e3344e0184b33a699b19a29866bc1e41201ace1a995e8ca3149']

Name

parquesanrafael.cl

Pattern Type

stix

Pattern

[domain-name:value = 'parquesanrafael.cl']

Name

gavice.ng

Pattern Type

stix

Pattern

[domain-name:value = 'gavice.ng']

Name

resetlocations.com

Pattern Type

stix

Pattern

[domain-name:value = 'resetlocations.com']

Name

c62199ef9c2736d15255f5deaa663158a7bb3615ba9262eb67e3f4adada14111

Description

SHA256 of 0032b8eabdc41e01923fabca5fe8a06b

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'c62199ef9c2736d15255f5deaa663158a7bb3615ba9262eb67e3f4adada14111']

Name

0c6066c71fef8c472b98b4dc42b98b2f5302532d

Description

Detects the deobfuscation algorithm and rc4 from STATICNOISE

Pattern Type

yara

Pattern

rule M_Downloader_STATICNOISE_1 { meta: author = "Mandiant" date_created = "2023-04-14" description = "Detects the deobfuscation algorithm and rc4 from STATICNOISE" version = "1" weight = "100" strings: $ = {41 8A C8 48 B8 [8] 80 E1 07 C0 E1 03 48 D3 E8 41 30 04 10 49 FF C0} $ = {80 E1 07 C0 E1 03 48 b8 [8] 48 D3 E8 30 04 17 48 FF C7 48 83 FF} $ = {40 88 2C 3A 49 8B 02 88 0C 06 45 89 0B 44 89 03 4D 8B 0A} $ = {4D 8B 0A 46 0F BE 04 0A 44 03 C1 41 81 E0 FF 00 00 80} condition: all of them }

Name

59e5b2a7a3903e4fb9a23174b655adb75eb490625ddb126ef29446e47de4099f

Description

SHA256 of 301a7273418bceaa3fb15b15f69dd32a

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'59e5b2a7a3903e4fb9a23174b655adb75eb490625ddb126ef29446e47de4099f']

Name

simplesalsamix.com

Pattern Type

stix

Pattern

[domain-name:value = 'simplesalsamix.com']

Name

3157669431e690b2859c67bc99068f14f07be39b

Description

Detects the structure of the Donut loader

Pattern Type

yara

Pattern

rule M_Dropper_Donut_1 { meta: author = "Mandiant" date_created = "2023-04-12" description = "Detects the structure of the Donut loader" version = "1" weight = "100" condition: uint8(0) == 0xE8 and uint32(1) == uint32(5) and uint8(uint32(1)+5) == 0x59 }

Name

www.willyminiatures.com

Pattern Type

stix

Pattern

[hostname:value = 'www.willyminiatures.com']

Name

30fdf6337a01168eaa7d68a1bc4e5aa32faf9c23

Description

Detects the RC4 encryption algorithm used in MUSKYBEAT

Pattern Type

yara

Pattern

rule M_Dropper_MUSKYBEAT_1 { meta: author = "Mandiant" date_created = "2023-04-06" description = "Detects the RC4 encryption algorithm used in MUSKYBEAT" version = "1" weight = "100" disclaimer = "This rule is meant for hunting and is not tested to run in a production environment." strings: $ = {42 8A 14 04 48 8D ?? ?? ?? ?? ?? 8A C2 41 02 04 08 44 02 D0 41 0F B6 CA} $ = {41 B9 04 00 00 00 41 B8 00 30 00 00 48 8B D3 33 C9} condition: all of them }

Name

kitaeri.com

Pattern Type

stix

Pattern

[domain-name:value = 'kitaeri.com']

Name

7ebbfde758b21b31ae20ee24856247a00e09635e

Description

Detects Shellcode RDI projects from https://github.com/monoxgas/sRDI/blob/master/ShellcodeRDI

Pattern Type

yara

Pattern

rule M_Hunting_DaveShell_Dropper_1_2 { meta: author = "Mandiant" description = "Detects Shellcode RDI projects from https://github.com/monoxgas/sRDI/blob/master/ShellcodeRDI" disclaimer = "This rule is meant for hunting and is not tested to run in a production environment." strings: $ep = {E8 00 00 00 00 59 49 89 C8 BA [4] 49 81 c0 [4] 41 b9 [4] 56 48 89 e6 48 83 ?? f0 48 83 ec 30 48 89 4c 24 ?? 48 81 c1 [4] c7 44 24 ?? [4] e8} condition: $ep at 0 }

Name

2fc6dd4a248d402145d3a631764570e1da18f4ea

Description

Searches for the custom chaskey implementation

Pattern Type

yara

Pattern

rule M_Dropper_BURNTBATTER_1 { meta: author = "Mandiant" date_created = "2023/04/26" description = "Searches for the custom chaskey implementation" version = "1" weight = "100" disclaimer = "This rule is meant for hunting and is not tested to run in a production environment." strings: $chaskey_imp = {41 81 C8 20 20 20 20 41 81 F8 6B 65 72 6E} condition: any of them }

Name

311e9c8cf6d0b295074ffefaa9f277cb1f806343be262c59f88fbdf6fe242517

Description

SHA256 of 556857ccb27b527e05415eb6d443aee1

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '311e9c8cf6d0b295074ffefaa9f277cb1f806343be262c59f88fbdf6fe242517']

Name

sgrhf.org.pk

Pattern Type

stix

Pattern

[domain-name:value = 'sgrhf.org.pk']

Name

60d96d8d3a09f822ded0a3c84194a5d88ed62a979cbb6378545b45b04353bb37

Description

SHA256 of 129da1e7c8613fd8c2843d9ec191e30e

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '60d96d8d3a09f822ded0a3c84194a5d88ed62a979cbb6378545b45b04353bb37']

Name

b6d26c5b2b2300fa8bf784919638ba849805896cf969c5c330668b350907c148

Description

invalid_trailer_structure SHA256 of 50f57a4a4bf2c4b504954a36d48c99e7

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'b6d26c5b2b2300fa8bf784919638ba849805896cf969c5c330668b350907c148']

Name

62ce8e1489a8b87539792c07179faf1db1b46caa39b55902a4d82dcec44d72ae

Description

SHA256 of 62b2031f8988105efdf473bdfedd07f5

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '62ce8e1489a8b87539792c07179faf1db1b46caa39b55902a4d82dcec44d72ae']

Name

sylvio.com.br

Pattern Type

stix

Pattern

[domain-name:value = 'sylvio.com.br']

Name

0dd55a234be8e3e07b0eb19f47abe594295889564ce6a9f6e8cc4d3997018839

Description

SHA256 of 854e5c592e93b69b8ab08dbc8a0b673f

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '0dd55a234be8e3e07b0eb19f47abe594295889564ce6a9f6e8cc4d3997018839']

Name

6c55195f025fb895f9d0ec3edbf58bc0aa46c43eeb246cfb88eef1ae051171b3

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '6c55195f025fb895f9d0ec3edbf58bc0aa46c43eeb246cfb88eef1ae051171b3']

Name

19442634bc2e0bfa6d08b7be333a351b932a517a1002c0e1c49fea8381372a6e

Description

invalid_trailer_structure SHA256 of dfbdd308e22898f680b6c2c8eb052fb5

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = '19442634bc2e0bfa6d08b7be333a351b932a517a1002c0e1c49fea8381372a6e']

Name

ae79aa17e6f3cc8e816e32335738b61b343e78c20abb8ae044adfeac5d97bf70

Description

SHA256 of 0be11b4f34ede748892ea49e473d82db

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'ae79aa17e6f3cc8e816e32335738b61b343e78c20abb8ae044adfeac5d97bf70']

Name

7fc9e830756e23aa4b050f4ceaeb2a83cd71cfc0145392a0bc03037af373066b

Description

SHA256 of 5e1389b494edc86e17ff1783ed6b9d37

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'7fc9e830756e23aa4b050f4ceaeb2a83cd71cfc0145392a0bc03037af373066b']

Name

7a9d27006887464220c456cc1cdbcf7766bc8fd760114b79b04a7e3fef73b33a

Description

SHA256 of f4ef5672af889429d95f111ea65ff490

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' =
'7a9d27006887464220c456cc1cdbcf7766bc8fd760114b79b04a7e3fef73b33a']

Name

sharpledge.com

Description

QUARTERRIG C2 Domain

Pattern Type

stix

Pattern

[domain-name:value = 'sharpledge.com']

Name

d7bda5e39327fe12b0c1f42c8e27787f177a352f8eebafbe35d3e790724eceff

Description

SHA256 of b48a16fdf890283cac7484ef0911a1f2

Pattern Type

stix

Pattern

[file:hashes.'SHA-256' = 'd7bda5e39327fe12b0c1f42c8e27787f177a352f8eebafbe35d3e790724eceff']

Intrusion-Set

Name
APT29

Description
[APT29](https://attack.mitre.org/groups/G0016) is threat group that has been attributed to Russia's Foreign Intelligence Service (SVR).(Citation: White House Imposing Costs RU Gov April 2021)(Citation: UK Gov Malign RIS Activity April 2021) They have operated since at least 2008, often targeting government networks in Europe and NATO member countries, research institutes, and think tanks. [APT29](https://attack.mitre.org/groups/G0016) reportedly compromised the Democratic National Committee starting in the summer of 2015.(Citation: F-Secure The Dukes)(Citation: GRIZZLY STEPPE JAR)(Citation: Crowdstrike DNC June 2016)(Citation: UK Gov UK Exposes Russia SolarWinds April 2021) In April 2021, the US and UK governments attributed the [SolarWinds Compromise](https://attack.mitre.org/campaigns/C0024) to the SVR; public statements included citations to [APT29](https://attack.mitre.org/groups/G0016), Cozy Bear, and The Dukes.(Citation: NSA Joint Advisory SVR SolarWinds April 2021)(Citation: UK NSCS Russia SolarWinds April 2021) Industry reporting also referred to the actors involved in this campaign as UNC2452, NOBELIUM, StellarParticle, Dark Halo, and SolarStorm.(Citation: FireEye SUNBURST Backdoor December 2020)(Citation: MSTIC NOBELIUM Mar 2021)(Citation: CrowdStrike SUNSPOT Implant January 2021)(Citation: Volexity SolarWinds)(Citation: Cybersecurity Advisory SVR TTP May 2021)(Citation: Unit 42 SolarStorm December 2020)

Country

Name
Czechia

Name
Ukraine

Domain-Name

Value
kitaeri.com
kegas.id
sharpledge.com
resetlocations.com
sgrhf.org.pk
gavice.ng
sgrfh.org.pk
sylvio.com.br
inovaoftalmologia.com.br
parquesanrafael.cl
simplesalsamix.com

StixFile

Value
19442634bc2e0bfa6d08b7be333a351b932a517a1002c0e1c49fea8381372a6e
ae79aa17e6f3cc8e816e32335738b61b343e78c20abb8ae044adfeac5d97bf70
6c55195f025fb895f9d0ec3edbf58bc0aa46c43eeb246cfb88eef1ae051171b3
311e9c8cf6d0b295074ffefaa9f277cb1f806343be262c59f88fbdf6fe242517
302c0d553c9e7f2561864d79022b780a53ec0a5927e8962d883b88dde249d044
7a9d27006887464220c456cc1cdbcf7766bc8fd760114b79b04a7e3fef73b33a
0dd55a234be8e3e07b0eb19f47abe594295889564ce6a9f6e8cc4d3997018839
60d96d8d3a09f822ded0a3c84194a5d88ed62a979cbb6378545b45b04353bb37
d7bda5e39327fe12b0c1f42c8e27787f177a352f8eebafbe35d3e790724eceff
b6d26c5b2b2300fa8bf784919638ba849805896cf969c5c330668b350907c148
7fc9e830756e23aa4b050f4ceaeb2a83cd71cfc0145392a0bc03037af373066b
59e5b2a7a3903e4fb9a23174b655adb75eb490625ddb126ef29446e47de4099f
38f8b8036ed2a0b5abb8fbf264ee6fd2b82dcd917f60d9f1d8f18d07c26b1534

a8ae10b43cbf4e3344e0184b33a699b19a29866bc1e41201ace1a995e8ca3149

c62199ef9c2736d15255f5deaa663158a7bb3615ba9262eb67e3f4adada14111

62ce8e1489a8b87539792c07179faf1db1b46caa39b55902a4d82dcec44d72ae

Hostname

Value
www.willyminiatures.com

External References

- https://otx.alienvault.com/pulse/6511f107da5fed8d065d9477
- https://www.mandiant.com/resources/blog/apt29-evolving-diplomatic-phishing