OST
Ostschweizer
Fachhochschule

Velociraptor Introduction

Enterprise Response Tooling

Cyrill Brunschwiler

11. Oktober 2024

Cyber Defense

Velociraptor ?

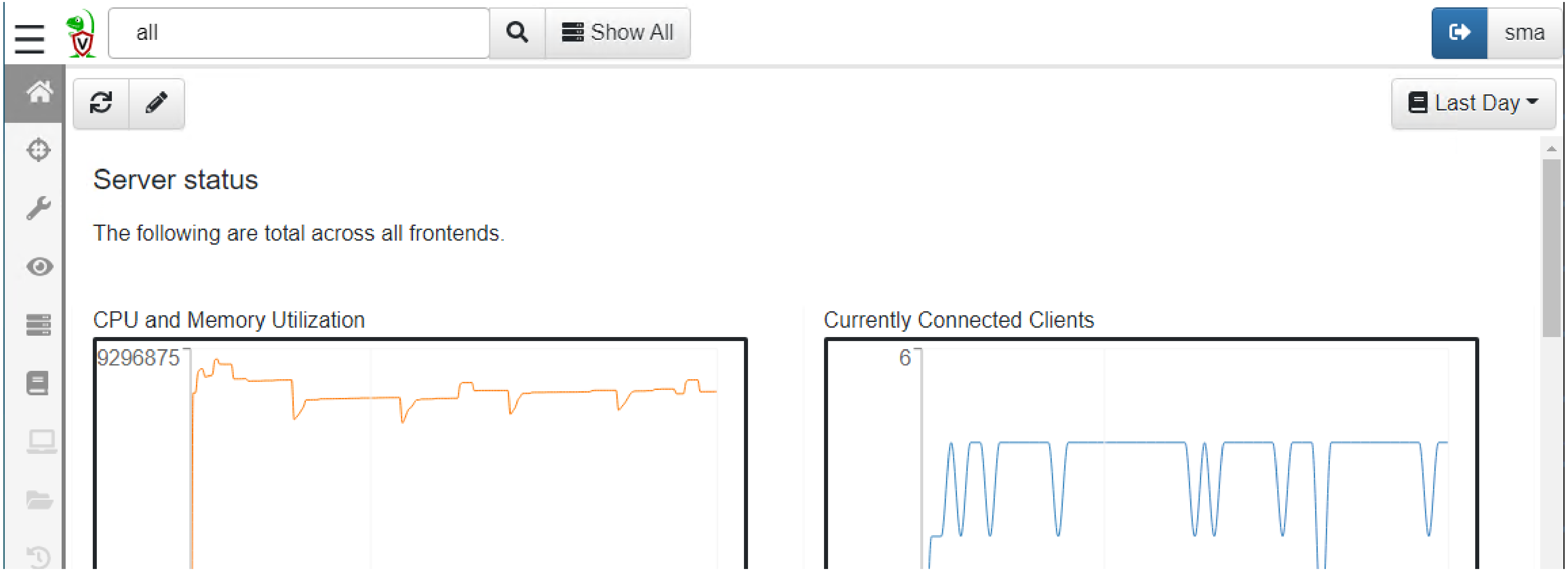
Credits: Universal Pictures

Velociraptor

Credits: Fred Wierum

Velociraptor !

Use Velociraptor to collect evidence, hunt for IOCs or just to monitor for something to happen. For that purpose, there are different kinds of client and server artifacts (scripts to collect stuff) some general and some that trigger on events which you could use for monitoring.



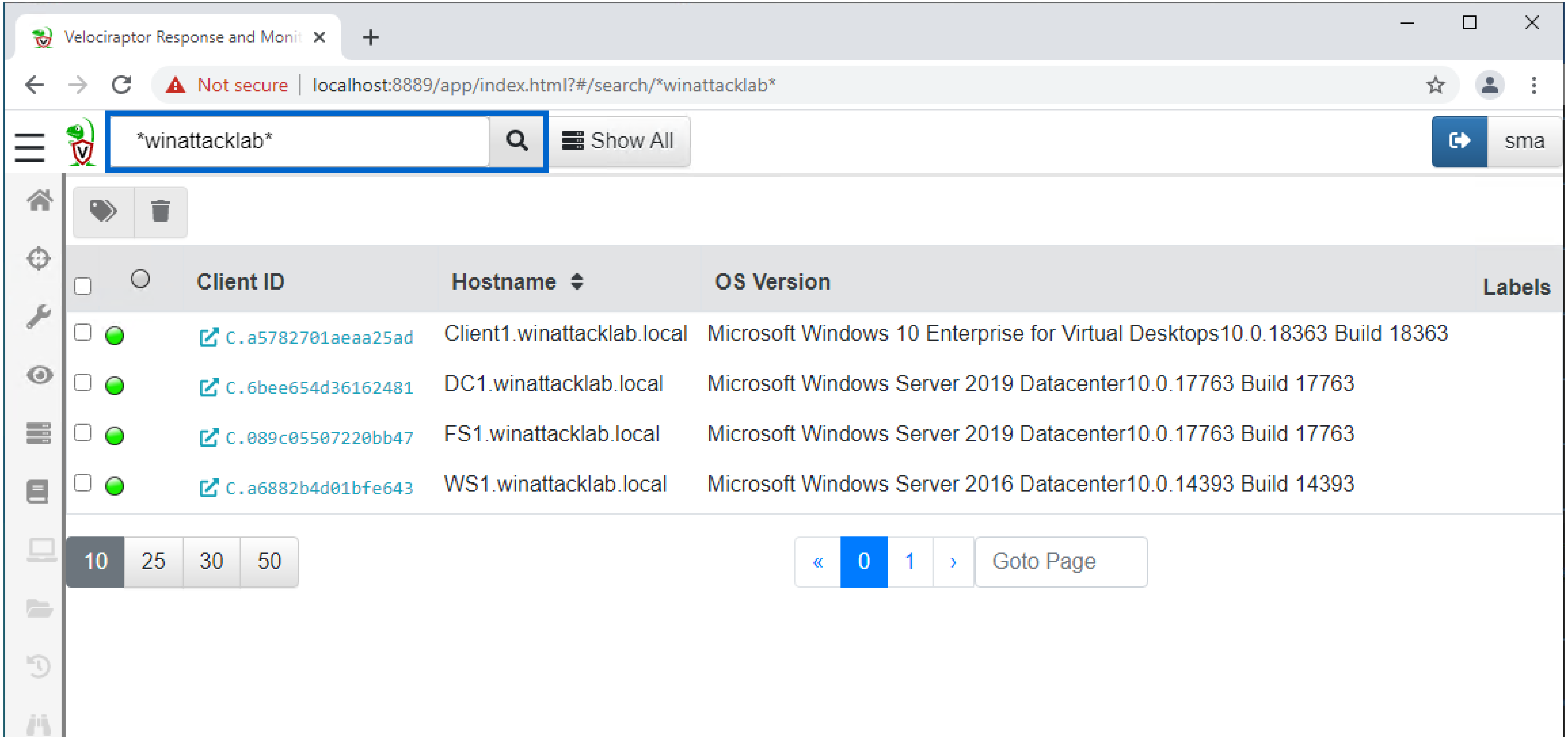
Server status

The following are total across all frontends.

Usually, connected clients stats curve reflects very much the general working hours

User Interface Basics

Search Connected Clients

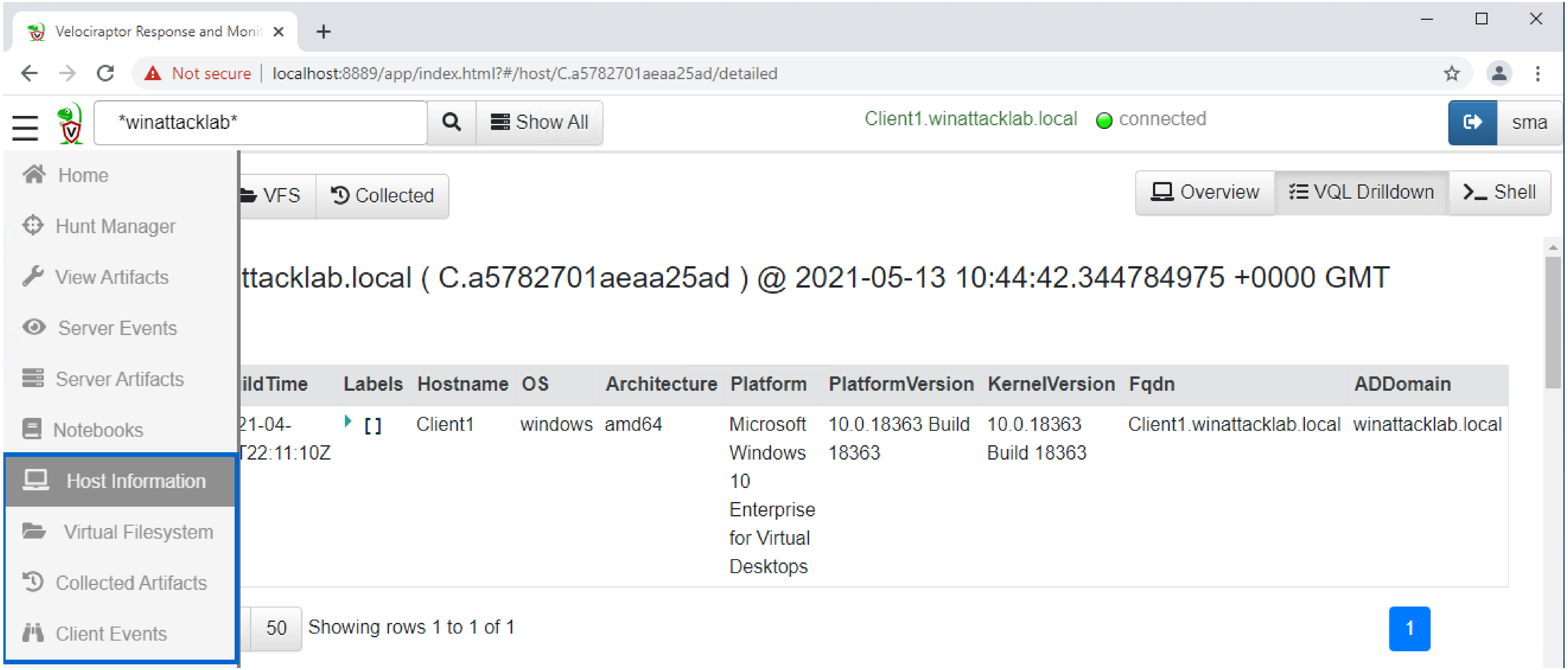


Velociraptor Response and Monit
Not secure
localhost:8889/app/index.html?#/search/*winattacklab*
winattacklab
Show All
sma
10
25
30
50
«
0
1
›
Goto Page

□	○	Client ID	Hostname	OS Version	Labels
□	●	C.a5782701aeaa25ad	Client1.winattacklab.local	Microsoft Windows 10 Enterprise for Virtual Desktops10.0.18363 Build 18363	
□	●	C.6bee654d36162481	DC1.winattacklab.local	Microsoft Windows Server 2019 Datacenter10.0.17763 Build 17763	
□	●	C.089c05507220bb47	FS1.winattacklab.local	Microsoft Windows Server 2019 Datacenter10.0.17763 Build 17763	
□	●	C.a6882b4d01bfe643	WS1.winattacklab.local	Microsoft Windows Server 2016 Datacenter10.0.14393 Build 14393	

Note, for the lab, the forensics client is connected, too.

Access Connected Client



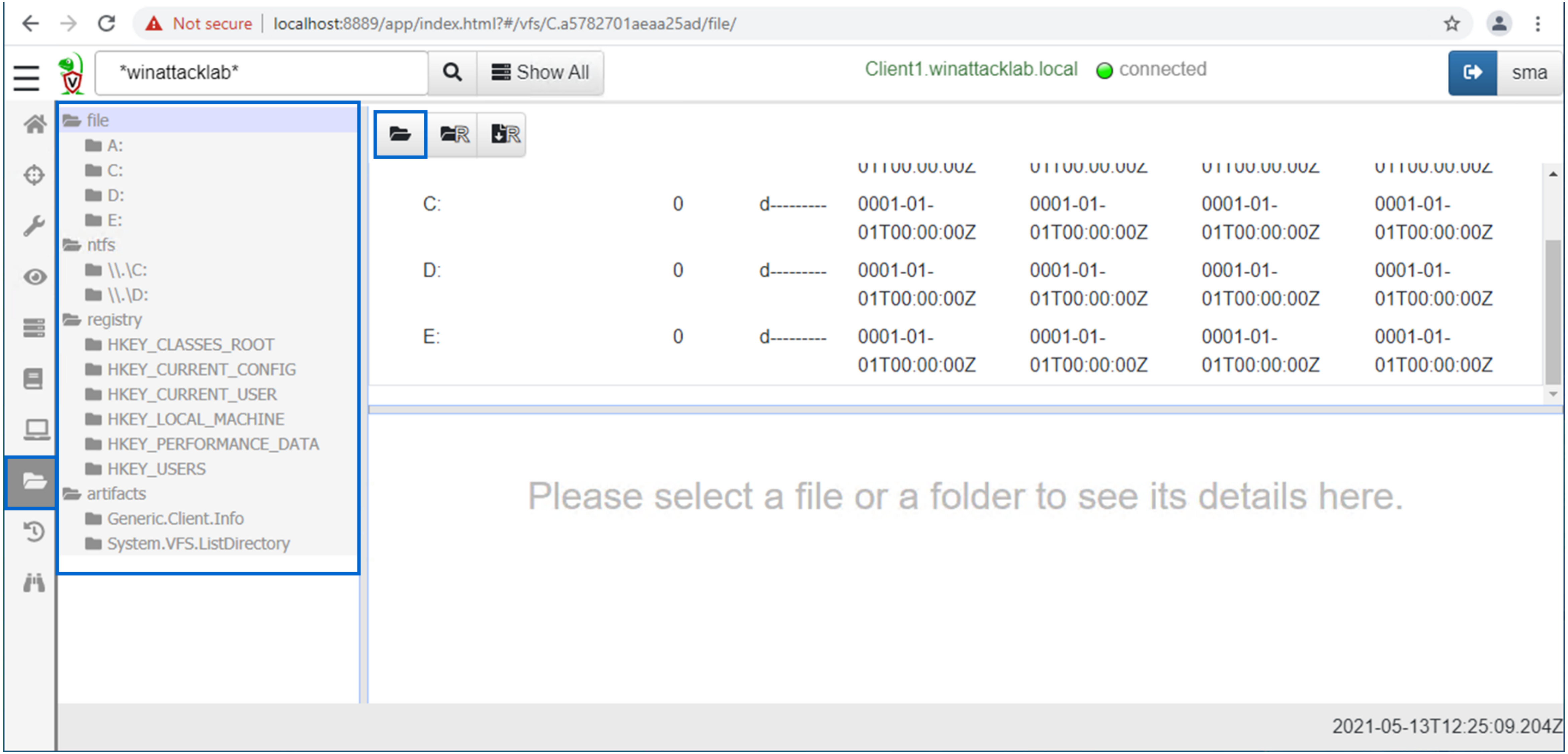
ttacklab.local (C.a5782701aeaa25ad) @ 2021-05-13 10:44:42.344784975 +0000 GMT

ildTime	Labels	Hostname	OS	Architecture	Platform	PlatformVersion	KernelVersion	Fqdn	ADDomain
21-04- T22:11:10Z	[]	Client1	windows	amd64	Microsoft Windows 10 Enterprise for Virtual Desktops	10.0.18363 Build 18363	10.0.18363 Build 18363	Client1.winattacklab.local	winattacklab.local

Showing rows 1 to 1 of 1

Select a client and the lower part of the menu becomes active. It’s always client dependent

Virtual File System



localhost:8889/app/index.html?#/vfs/C.a5782701aeaa25ad/file/

winattacklab

Client1.winattacklab.local ● connected

- file
  - A:
  - C:
  - D:
  - E:
- ntfs
  - \\.\C:
  - \\.\D:
- registry
  - HKEY_CLASSES_ROOT
  - HKEY_CURRENT_CONFIG
  - HKEY_CURRENT_USER
  - HKEY_LOCAL_MACHINE
  - HKEY_PERFORMANCE_DATA
  - HKEY_USERS
- artifacts
  - Generic.Client.Info
  - System.VFS.ListDirectory

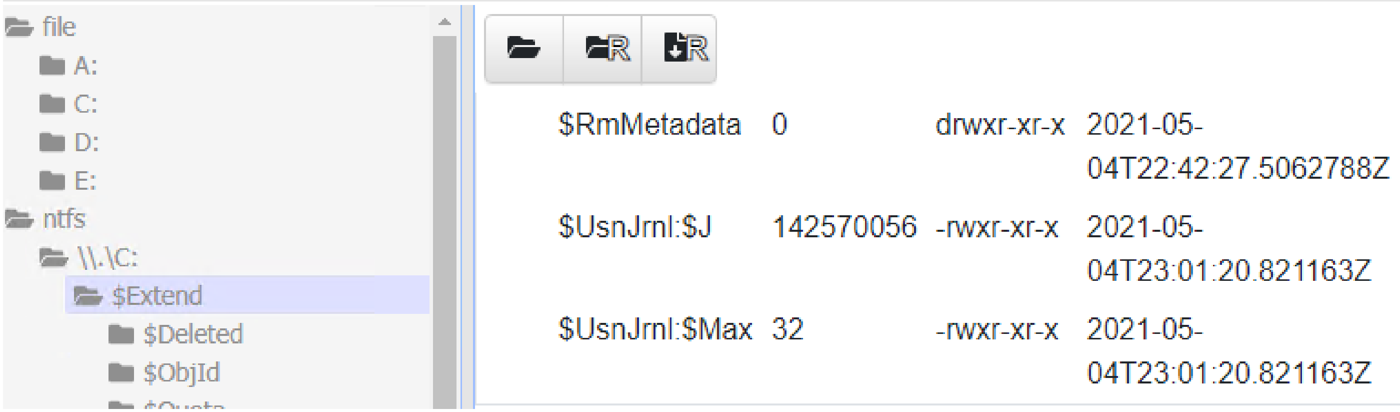
			01T00:00:00Z	01T00:00:00Z	01T00:00:00Z	01T00:00:00Z
C:	0	d---------	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z
D:	0	d---------	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z
E:	0	d---------	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z	0001-01-01T00:00:00Z

Please select a file or a folder to see its details here.

2021-05-13T12:25:09.204Z

Virtual File System (VFS)

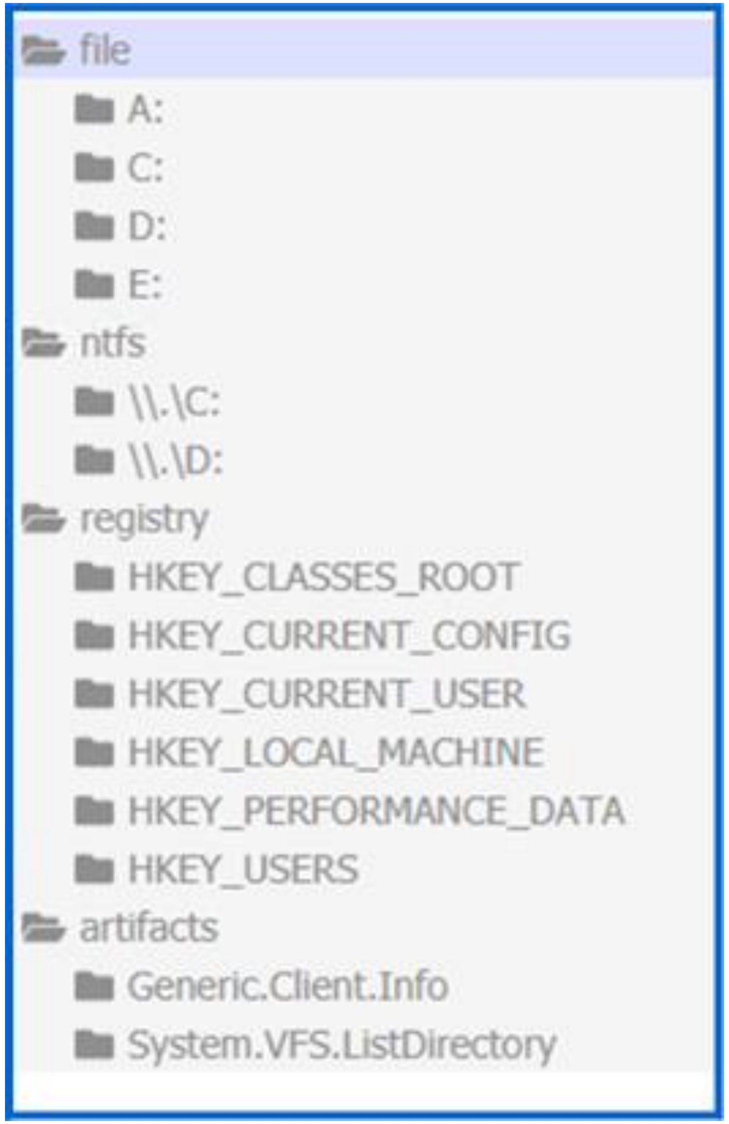
- **File** - File system access based on OS FS API
- **NTFS** – NTFS raw parsing filesystem access



```
file
  A:
  C:
  D:
  E:
ntfs
  \\.\C:
    $Extend
      $Deleted
      $ObjId
```

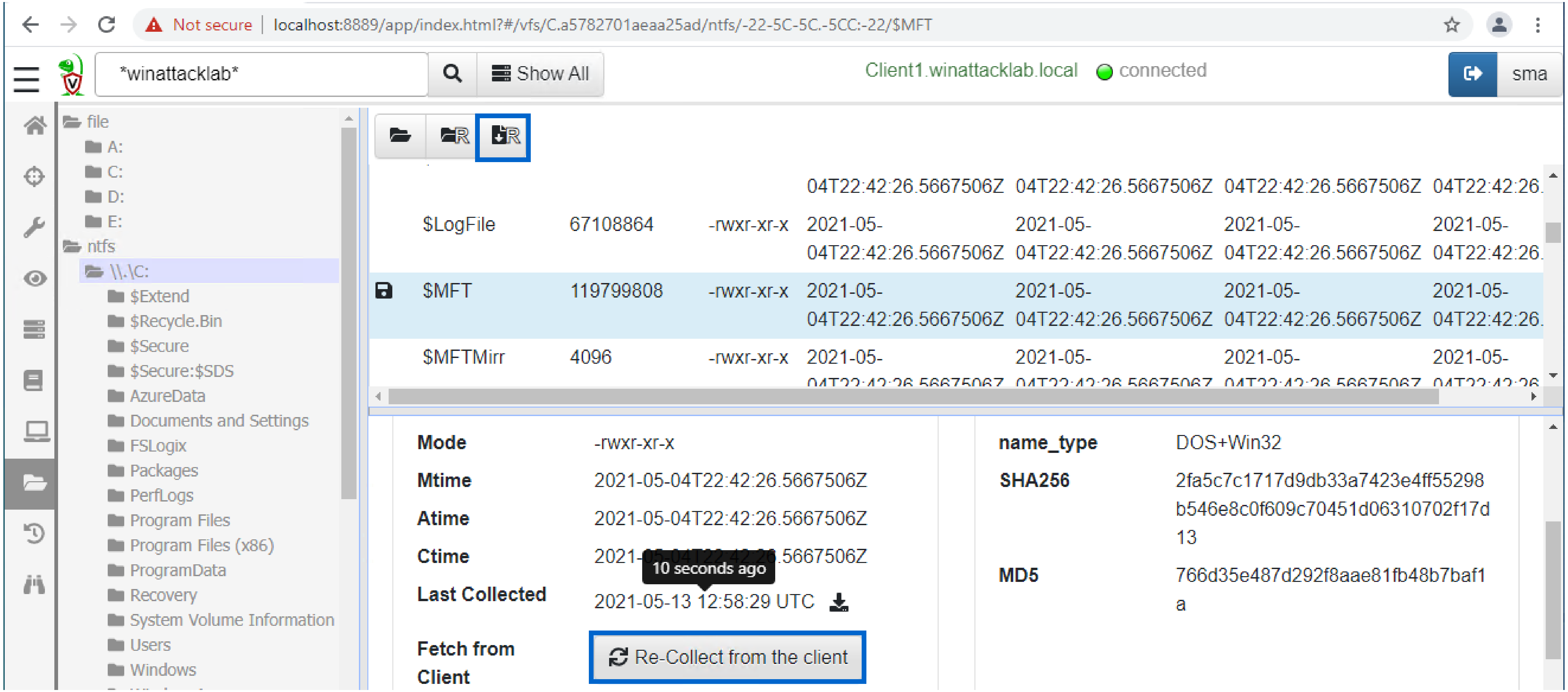
```
$RmMetadata   0          drwxr-xr-x  2021-05-
                                     04T22:42:27.5062788Z
$UsnJrnl:$J   142570056  -rwxr-xr-x  2021-05-
                                     04T23:01:20.821163Z
$UsnJrnl:$Max 32         -rwxr-xr-x  2021-05-
                                     04T23:01:20.821163Z
```

- **Registry** - Windows Registry access using the Registry API
- **Artifacts** - Artifacts collected from the client incl. type and time in Velociraptor Artifacts are commands and scripts that actually grab some data (we usually call these artifacts) from clients.



```
file
  A:
  C:
  D:
  E:
ntfs
  \\.\C:
  \\.\D:
registry
  HKEY_CLASSES_ROOT
  HKEY_CURRENT_CONFIG
  HKEY_CURRENT_USER
  HKEY_LOCAL_MACHINE
  HKEY_PERFORMANCE_DATA
  HKEY_USERS
artifacts
  Generic.Client.Info
  System.VFS.ListDirectory
```

File Download (collection)



			04T22:42:26.5667506Z	04T22:42:26.5667506Z	04T22:42:26.5667506Z	04T22:42:26.
$LogFile	67108864	-rwxr-xr-x	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.
$MFT	119799808	-rwxr-xr-x	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.5667506Z	2021-05-04T22:42:26.
$MFTMirr	4096	-rwxr-xr-x	2021-05-	2021-05-	2021-05-	2021-05-

Mode	-rwxr-xr-x
Mtime	2021-05-04T22:42:26.5667506Z
Atime	2021-05-04T22:42:26.5667506Z
Ctime	2021-05-04T22:42:26.5667506Z
Last Collected	2021-05-13 12:58:29 UTC
Fetch from Client	Re-Collect from the client

name_type	DOS+Win32
SHA256	2fa5c7c1717d9db33a7423e4ff55298b546e8c0f609c70451d06310702f17d13
MD5	766d35e487d292f8aae81fb48b7baf1a

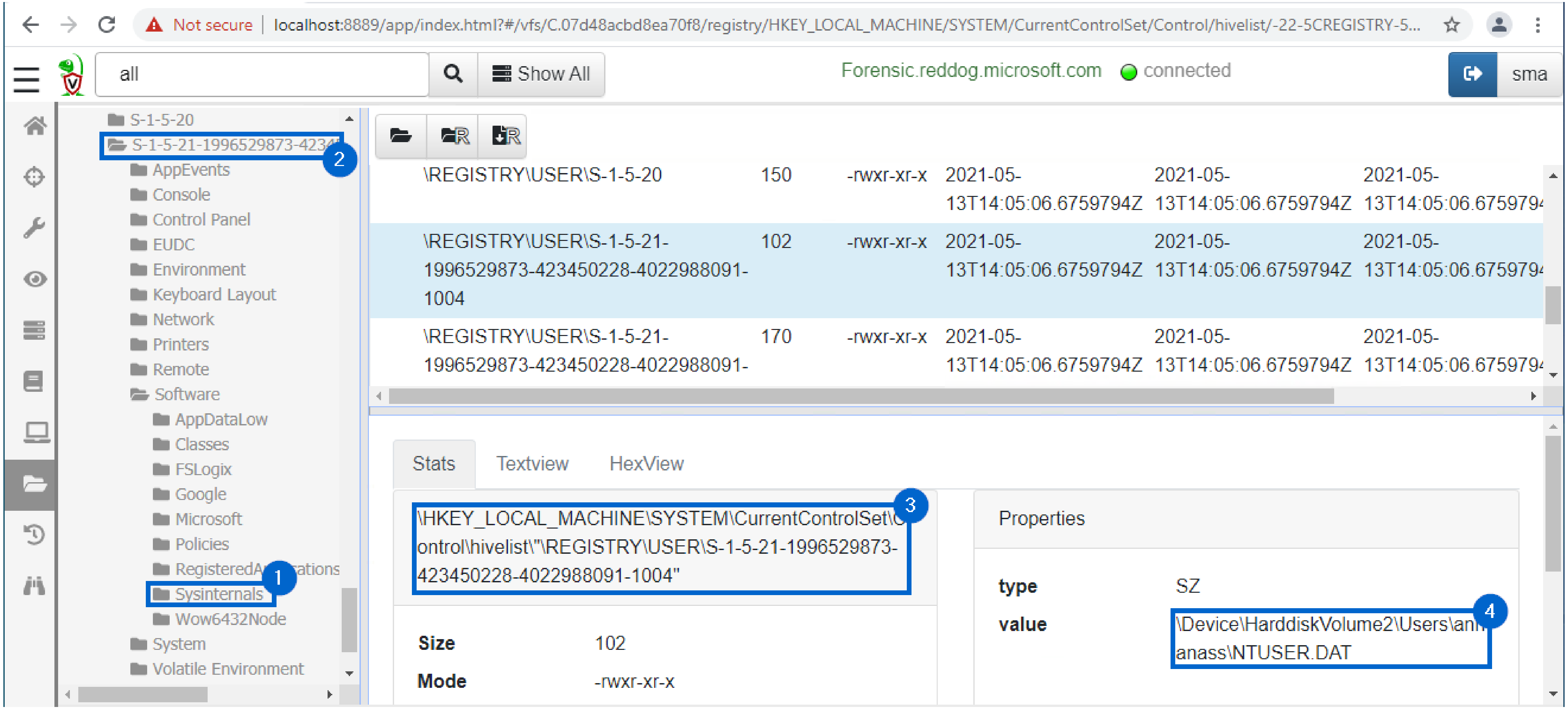
You may collect files individually (lower button) or an entire folder recursively (top button). Files get marked with the floppy once available on the server.

Velociraptor VFS Exercise

Find evidence of execution of Sysinternals tools on the Forensics machine using the VFS browser

As you may know, Sysinternals tools create a registry key when they're first run. Find out which Sysinternals tools have been run by users on the system.

Velociraptor VFS Exercise Solution



```
\REGISTRY\USER\S-1-5-20                     150   -rwxr-xr-x   2021-05-                   2021-05-                   2021-05-
                                                               13T14:05:06.6759794Z       13T14:05:06.6759794Z       13T14:05:06.675979
\REGISTRY\USER\S-1-5-21-                    102   -rwxr-xr-x   2021-05-                   2021-05-                   2021-05-
1996529873-423450228-4022988091-                               13T14:05:06.6759794Z       13T14:05:06.6759794Z       13T14:05:06.675979
1004
\REGISTRY\USER\S-1-5-21-                    170   -rwxr-xr-x   2021-05-                   2021-05-                   2021-05-
1996529873-423450228-4022988091-                               13T14:05:06.6759794Z       13T14:05:06.6759794Z       13T14:05:06.675979
```

\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\hivelist\"\REGISTRY\USER\S-1-5-21-1996529873-423450228-4022988091-1004"

Size 102

Mode -rwxr-xr-x

type SZ

value \Device\HarddiskVolume2\Users\annanass\NTUSER.DAT

User annanass executed some Sysinternals tools

Velociraptor Artifacts

Velociraptor Artifacts

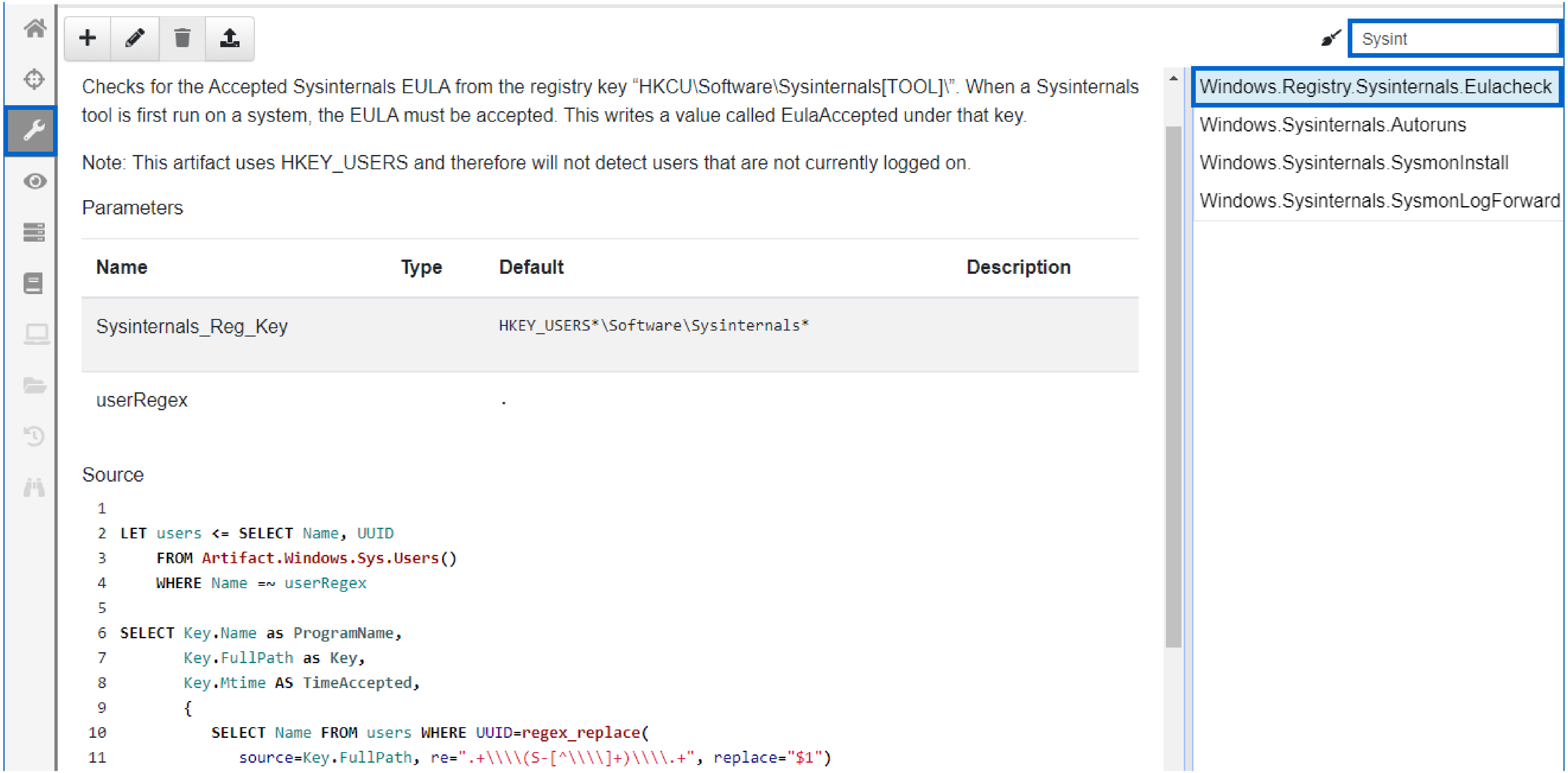
Velociraptor is a query language engine. Basically everything is a query (VQL).

Artifacts aim to encapsulate evidence collection

- Structure is YAML
- Ideally includes comments
- Allow for customization where reasonable
- Recursion. Use Artifacts in the query language

```
LET users <= SELECT Name, UUID
    FROM Artifact.Windows.Sys.Users()
    WHERE Name =~ userRegex

SELECT Key.Name as ProgramName,
       Key.FullPath as Key,
       Key.Mtime AS TimeAccepted,
       {
           SELECT Name FROM users WHERE UUID=regex_replace(
               source=Key.FullPath, re=".+\\\\(S-[^\\\\]+)\\\\.+", re
       } as User,
       EulaAccepted
FROM read_reg_key(globs=split(string=Sysinternals_Reg_Key, sep='
```

Velociraptor Artifact, Sysinternals Example


+
Sysint
Windows.Registry.Sysinternals.Eulacheck
Windows.Sysinternals.Autoruns
Windows.Sysinternals.SysmonInstall
Windows.Sysinternals.SysmonLogForward

Checks for the Accepted Sysinternals EULA from the registry key "HKCU\Software\Sysinternals[TOOL]\". When a Sysinternals tool is first run on a system, the EULA must be accepted. This writes a value called EulaAccepted under that key.

Note: This artifact uses HKEY_USERS and therefore will not detect users that are not currently logged on.

Parameters

Name	Type	Default	Description
Sysinternals_Reg_Key		HKEY_USERS*\Software\Sysinternals*	
userRegex		.	

Source

```

LET users <= SELECT Name, UUID
    FROM Artifact.Windows.Sys.Users()
    WHERE Name =~ userRegex

SELECT Key.Name as ProgramName,
       Key.FullPath as Key,
       Key.Mtime AS TimeAccepted,
       {
          SELECT Name FROM users WHERE UUID=regex_replace(
             source=Key.FullPath, re=".+\\\\(S-[^\\\\]+)\\\\.+", replace="$1")
```

Velociraptor Artifact Exercise

Find evidence of execution of Sysinternals tools on the Forensics machine using Velociraptor Client Artifacts.

Using an Artifact comes with the advantage to not necessarily know every evidence location by heart and helps to avoid the nifty little faults that often happen while under pressure.

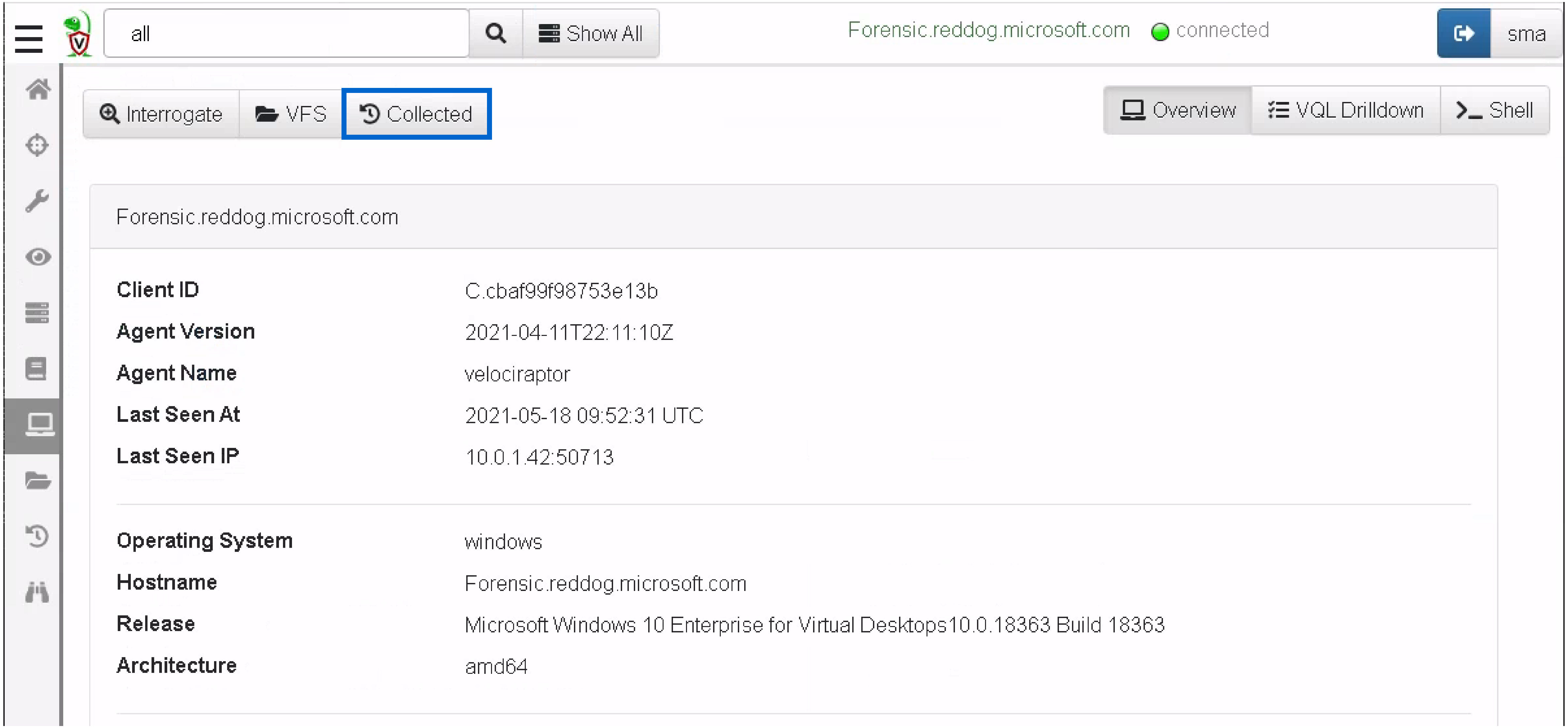
Artifacts FTW!

Search the Forensics machine for execution of Sysinternals tools and let me know

- Who (which user) ?
- What (which of the tools) ?
- When (the time of EULA acceptance) ?

Velociraptor Artifact Exercise Solution

Choose Client, Open Collected Items (Flows)

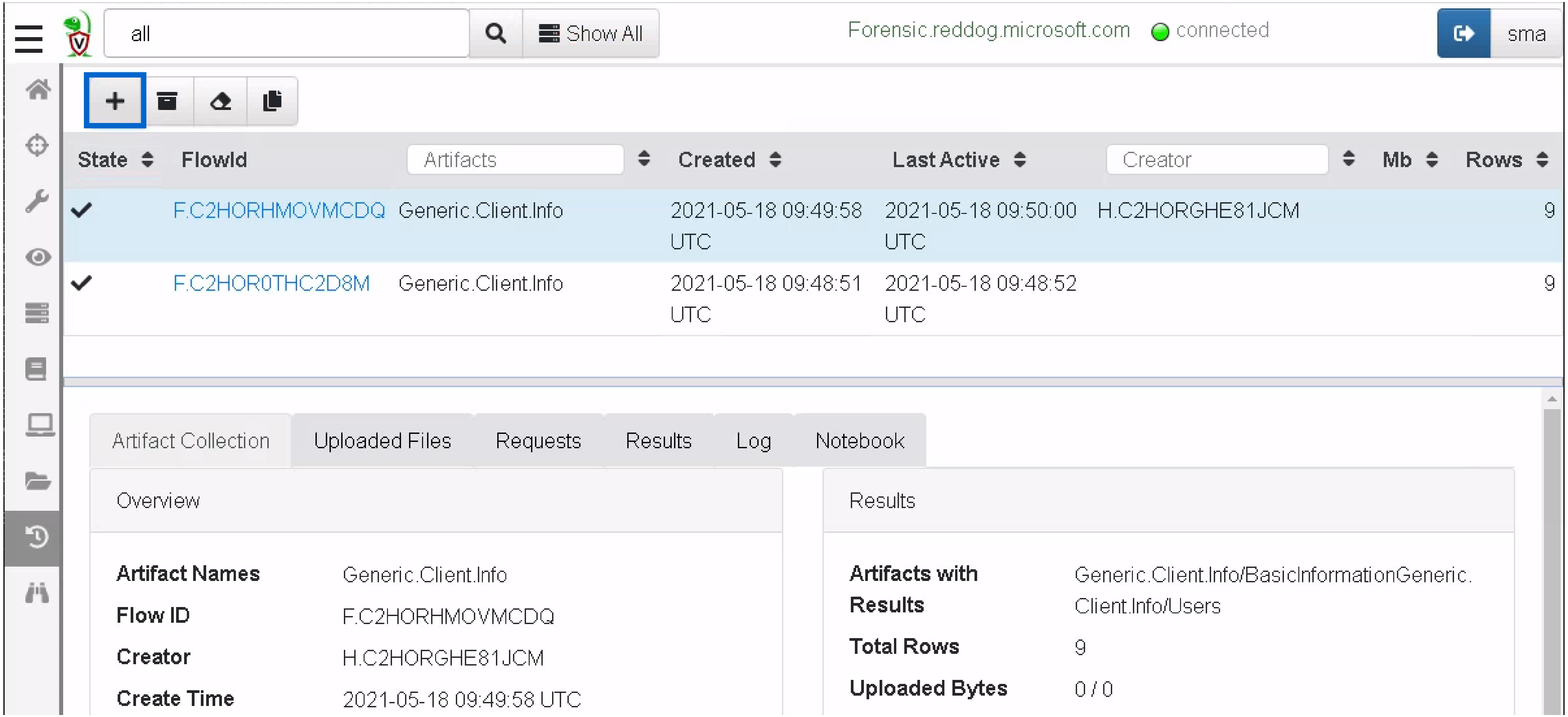


Forensic.reddog.microsoft.com

Client ID	C.cbaf99f98753e13b
Agent Version	2021-04-11T22:11:10Z
Agent Name	velociraptor
Last Seen At	2021-05-18 09:52:31 UTC
Last Seen IP	10.0.1.42:50713
Operating System	windows
Hostname	Forensic.reddog.microsoft.com
Release	Microsoft Windows 10 Enterprise for Virtual Desktops10.0.18363 Build 18363
Architecture	amd64

Velociraptor Artifact Exercise Solution

Add a New Flow



State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2HORHMOVMCDQ	Generic.Client.Info	2021-05-18 09:49:58 UTC	2021-05-18 09:50:00 UTC	H.C2HORGHE81JCM		9
✓	F.C2HOR0THC2D8M	Generic.Client.Info	2021-05-18 09:48:51 UTC	2021-05-18 09:48:52 UTC			9

Overview

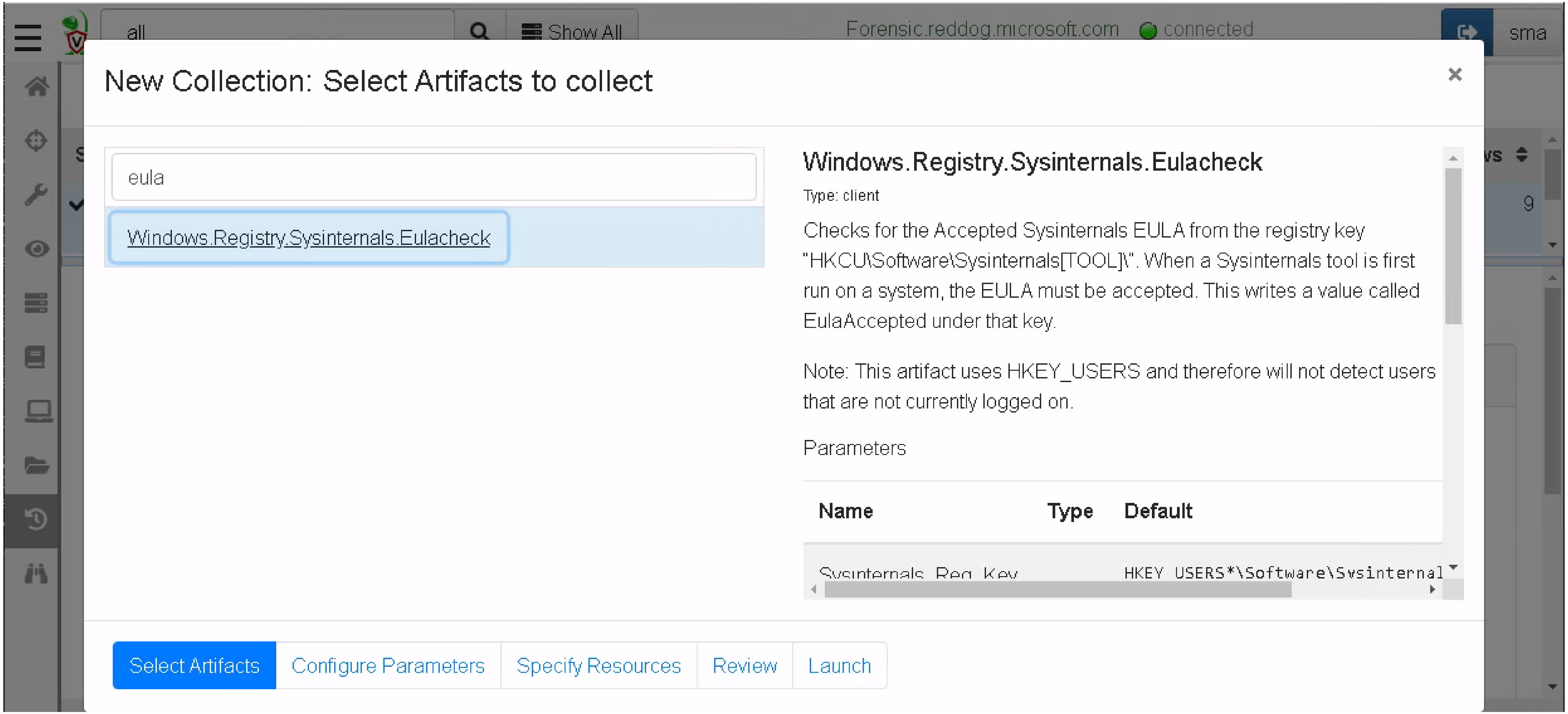
Artifact Names	Generic.Client.Info
Flow ID	F.C2HORHMOVMCDQ
Creator	H.C2HORGHE81JCM
Create Time	2021-05-18 09:49:58 UTC

Results

Artifacts with Results	Generic.Client.Info/BasicInformationGeneric.Client.Info/Users
Total Rows	9
Uploaded Bytes	0 / 0

Velociraptor Artifact Exercise Solution

Select Artifact, Configure, Specify Resources, Launch



New Collection: Select Artifacts to collect

eula

Windows.Registry.Sysinternals.Eulacheck

Windows.Registry.Sysinternals.Eulacheck

Type: client

Checks for the Accepted Sysinternals EULA from the registry key "HKCU\Software\Sysinternals[TOOL]\". When a Sysinternals tool is first run on a system, the EULA must be accepted. This writes a value called EulaAccepted under that key.

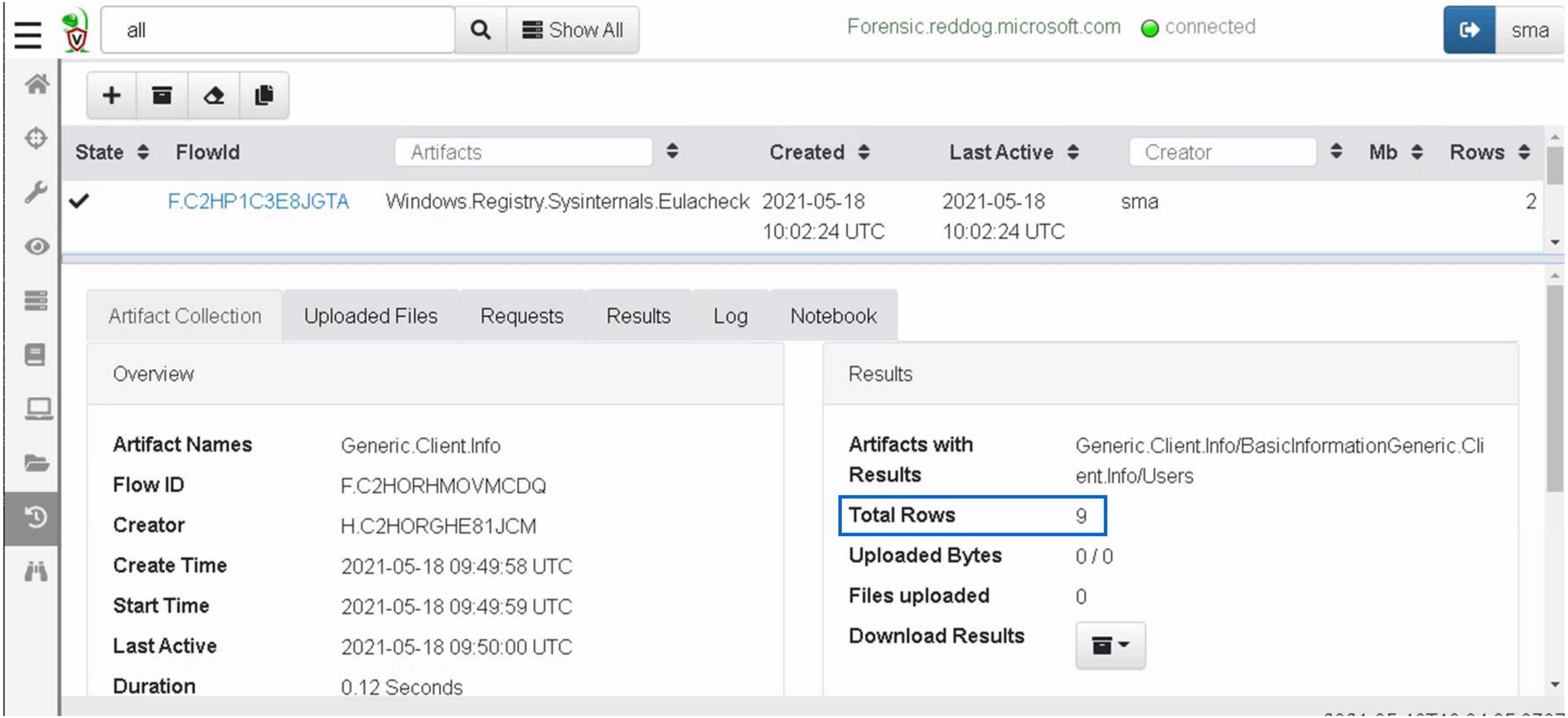
Note: This artifact uses HKEY_USERS and therefore will not detect users that are not currently logged on.

Parameters

Name	Type	Default
Sysinternals_Reg_Key		HKEY_USERS*\Software\Sysinternal

Velociraptor Artifact Exercise Solution

Check for Artifact Results



State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2HP1C3E8JGTA	Windows.Registry.Sysinternals.Eulacheck	2021-05-18 10:02:24 UTC	2021-05-18 10:02:24 UTC	sma		2

Overview

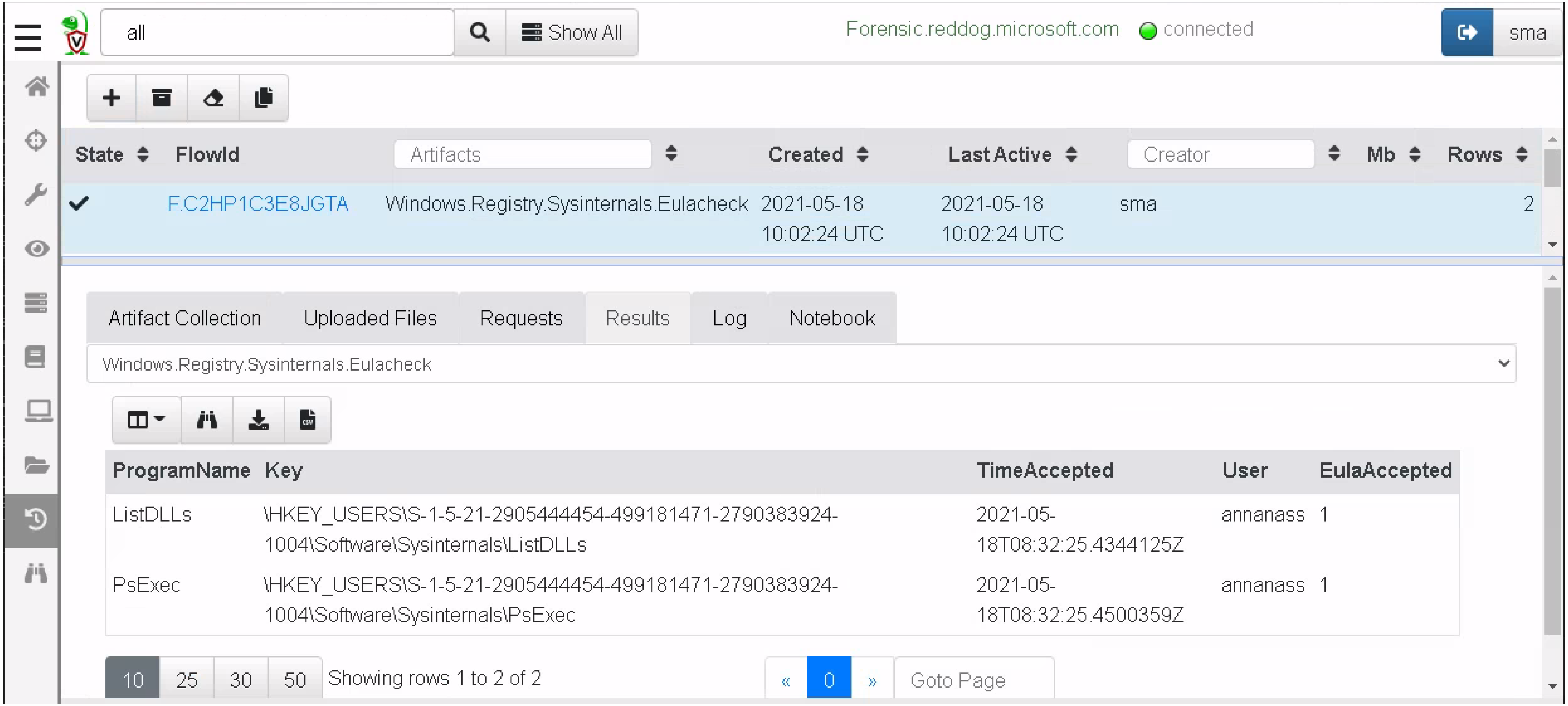
Artifact Names	Generic.Client.Info
Flow ID	F.C2HORHMOVMCDQ
Creator	H.C2HORGHE81JCM
Create Time	2021-05-18 09:49:58 UTC
Start Time	2021-05-18 09:49:59 UTC
Last Active	2021-05-18 09:50:00 UTC
Duration	0.12 Seconds

Results

Artifacts with Results	Generic.Client.Info/BasicInformationGeneric.Client.Info/Users
Total Rows	9
Uploaded Bytes	0 / 0
Files uploaded	0
Download Results	

Velociraptor Artifact Exercise Solution

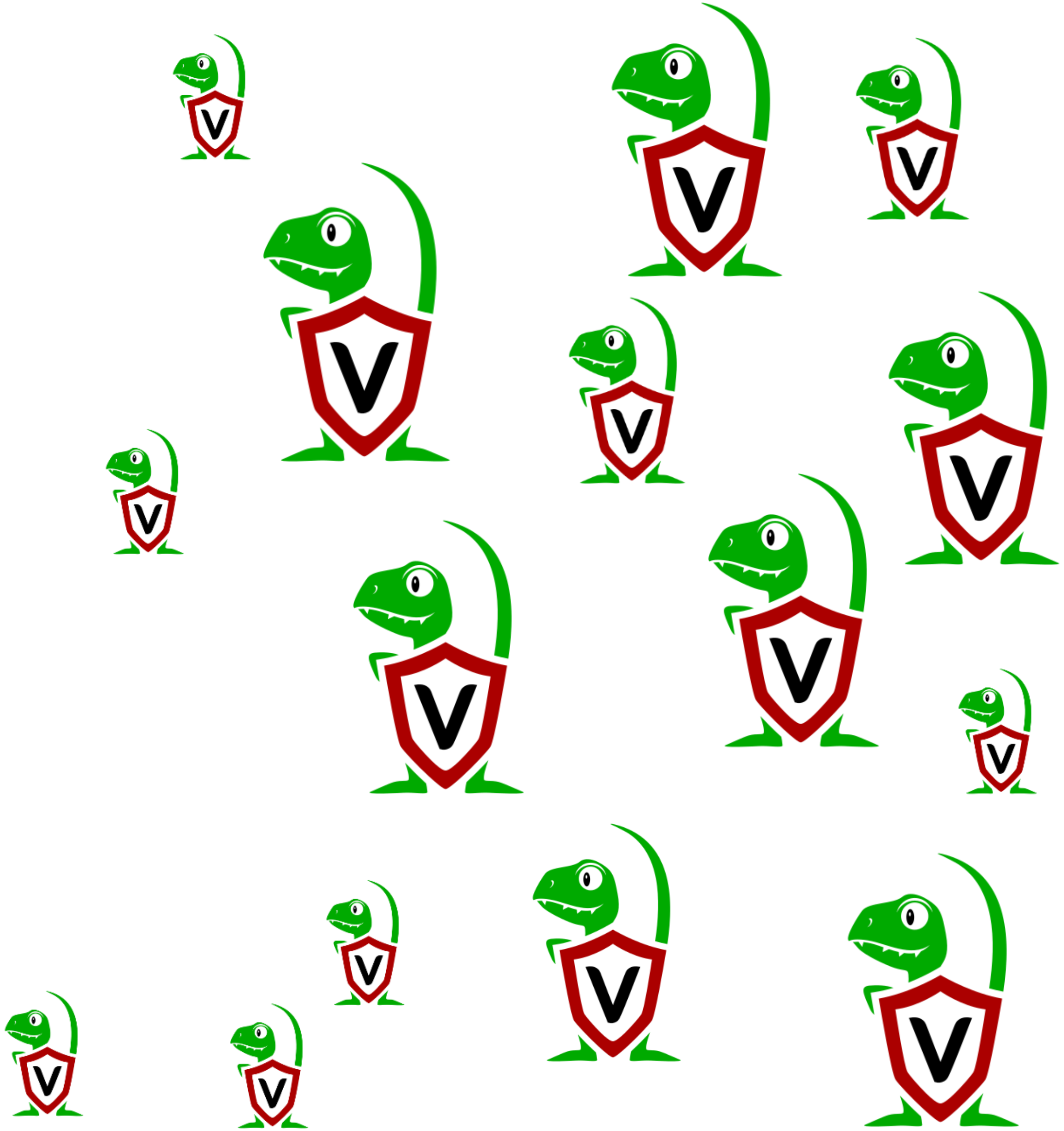
Switch to the Results Tab



State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2HP1C3E8JGTA	Windows.Registry.Sysinternals.Eulacheck	2021-05-18 10:02:24 UTC	2021-05-18 10:02:24 UTC	sma		2

ProgramName	Key	TimeAccepted	User	EulaAccepted
ListDLLs	\HKEY_USERS\S-1-5-21-2905444454-499181471-2790383924-1004\Software\Sysinternals\ListDLLs	2021-05-18T08:32:25.4344125Z	annanass	1
PsExec	\HKEY_USERS\S-1-5-21-2905444454-499181471-2790383924-1004\Software\Sysinternals\PsExec	2021-05-18T08:32:25.4500359Z	annanass	1

Velociraptor Hunts

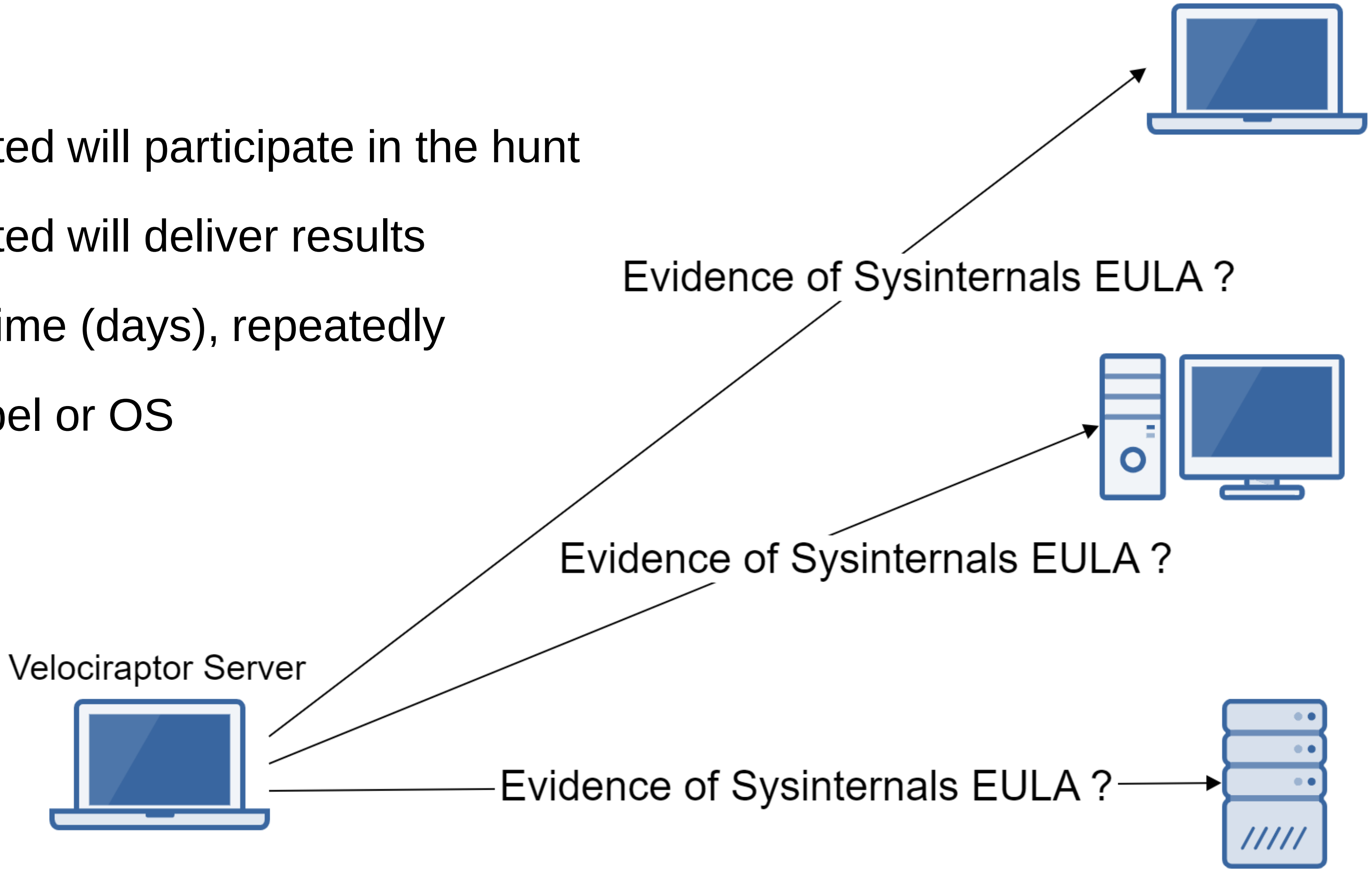


Velociraptor Hunting

Hunting enables you to collect the same artifacts over an entire fleet.

Note

- Only systems that are connected will participate in the hunt
- Only systems that are connected will deliver results
- Hunts are run for quite some time (days), repeatedly
- Hunts may be restricted by label or OS


Evidence of Sysinternals EULA ?
Evidence of Sysinternals EULA ?
Velociraptor Server
Evidence of Sysinternals EULA ?

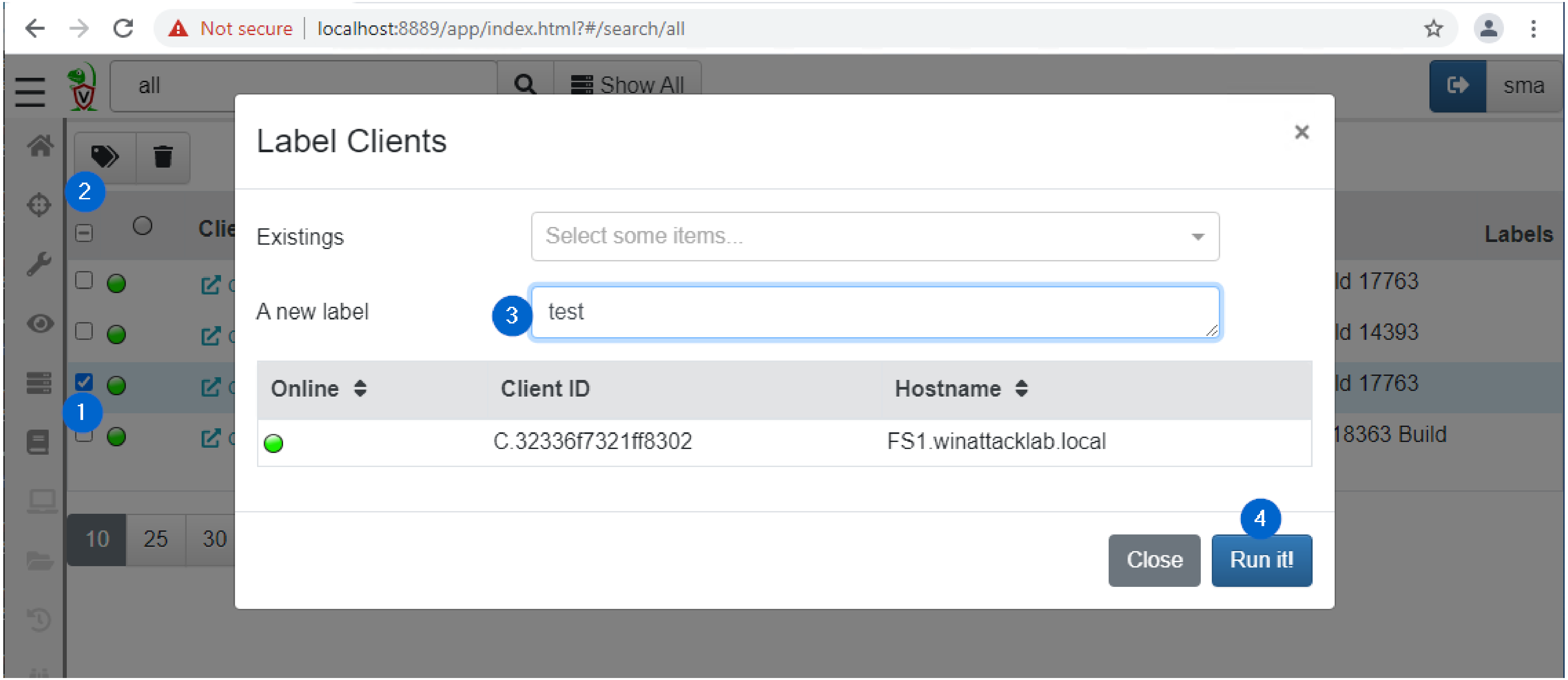
Velociraptor Hunt Exercise

Find evidence of execution of Sysinternals tools on labeled machines using a hunt

- Add a label “test” to the Forensics machine
- Run a hunt against “test” labeled systems using the Sysinternals EULA Artifact again

Velociraptor Hunt Exercise Solution

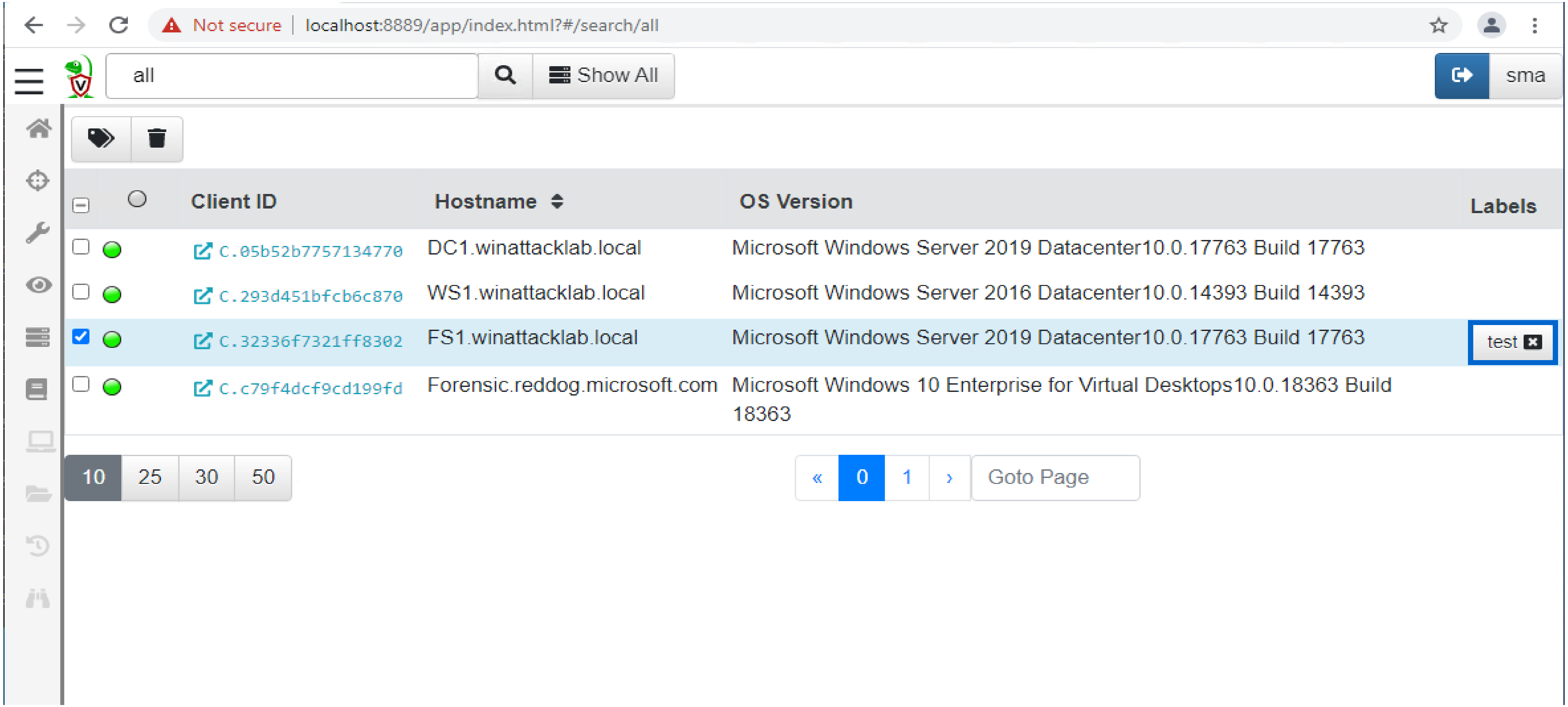
Add a Label to Selected Clients



Online	Client ID	Hostname
●	C.32336f7321ff8302	FS1.winattacklab.local

Velociraptor Hunt Exercise Solution

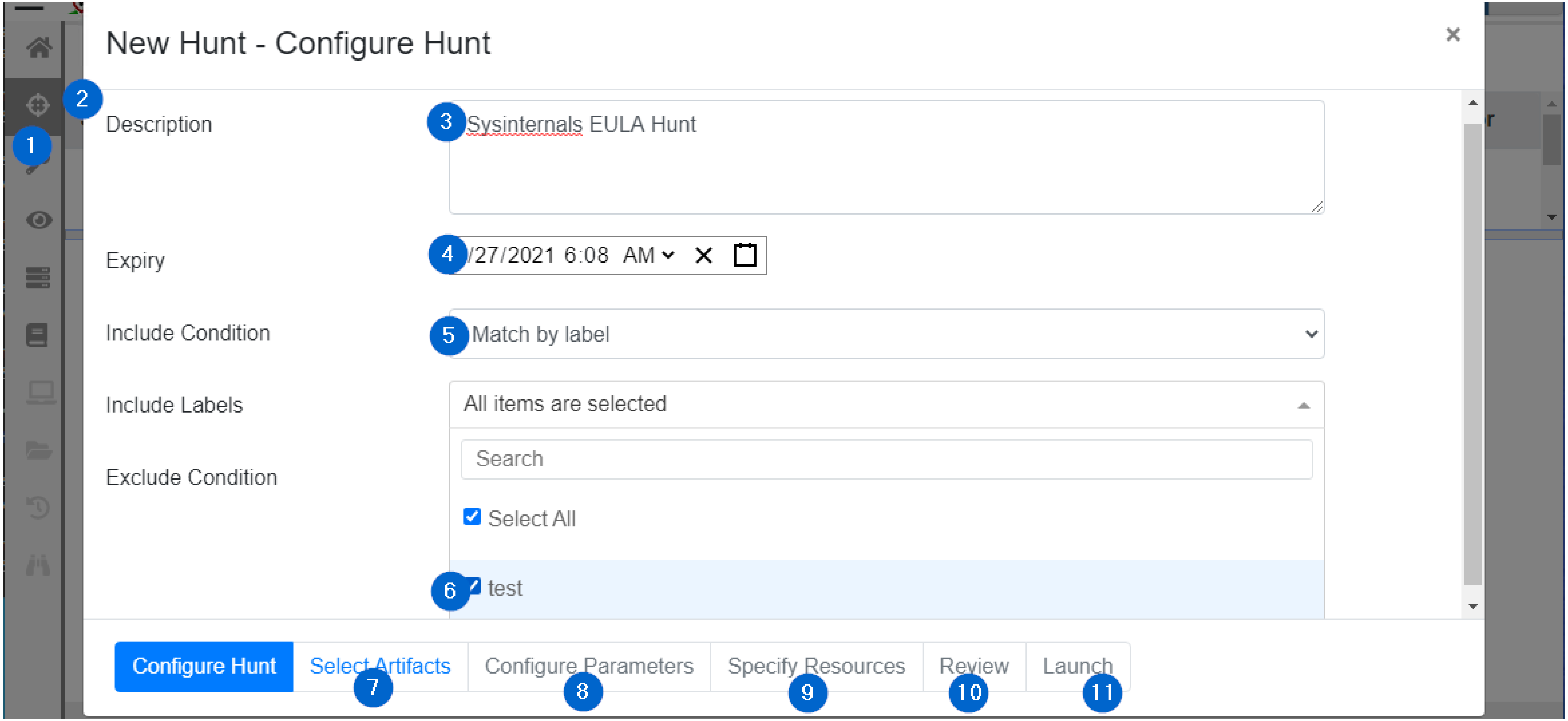
Forensics Machine Labeled with “test”



□	○	Client ID	Hostname	OS Version	Labels
□	●	C.05b52b7757134770	DC1.winattacklab.local	Microsoft Windows Server 2019 Datacenter10.0.17763 Build 17763	
□	●	C.293d451bfcb6c870	WS1.winattacklab.local	Microsoft Windows Server 2016 Datacenter10.0.14393 Build 14393	
☑	●	C.32336f7321ff8302	FS1.winattacklab.local	Microsoft Windows Server 2019 Datacenter10.0.17763 Build 17763	test ☒
□	●	C.c79f4dcf9cd199fd	Forensic.reddog.microsoft.com	Microsoft Windows 10 Enterprise for Virtual Desktops10.0.18363 Build 18363	

Velociraptor Hunt Exercise Solution

Add (+) a New Hunt on Systems with Label “test”



New Hunt - Configure Hunt

Description: Sysinternals EULA Hunt

Expiry: /27/2021 6:08 AM

Include Condition: Match by label

Include Labels: All items are selected

Search

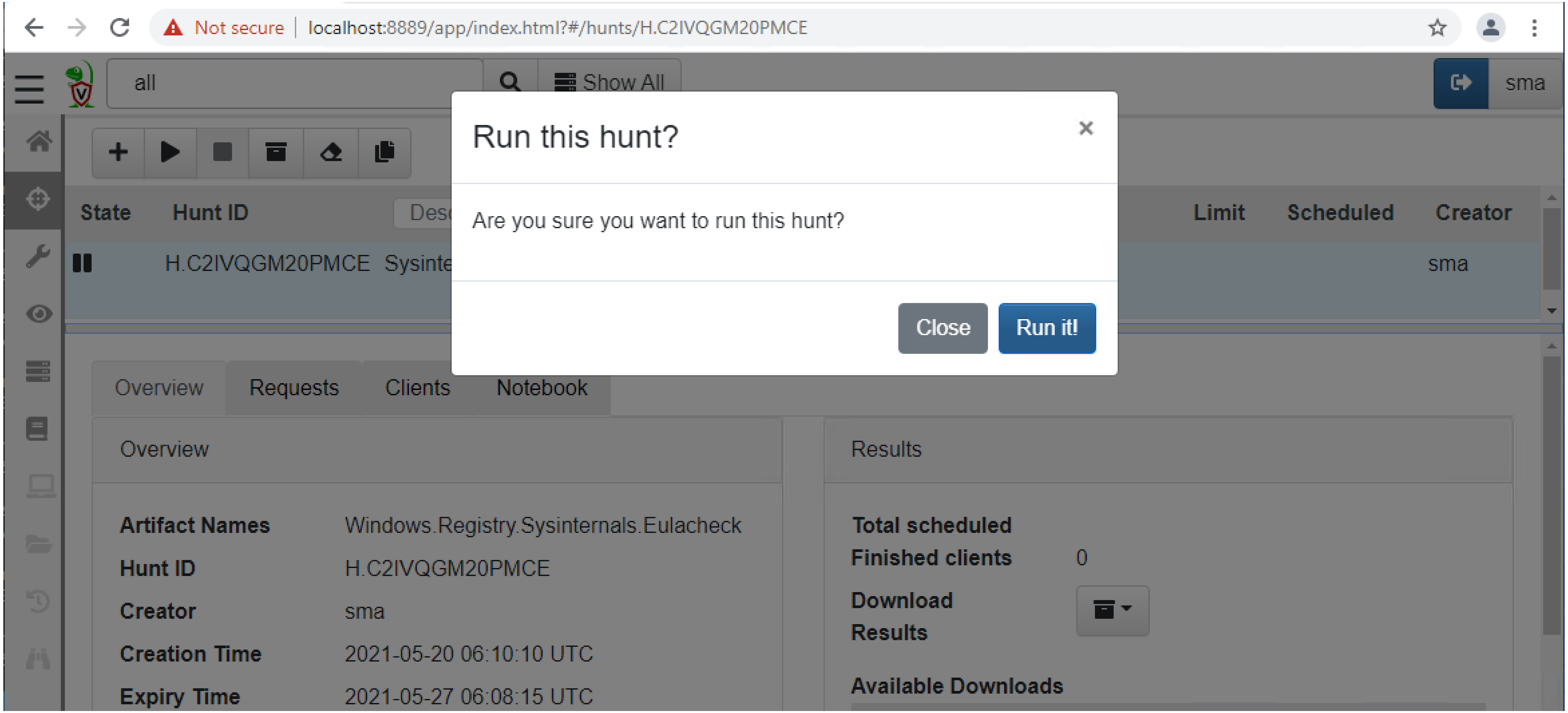
☑ Select All

☑ test

Exclude Condition

Velociraptor Hunt Exercise Solution

Newly Created Hunts are Always Paused, Run it



Run this hunt?

×

Are you sure you want to run this hunt?

Close

Run it!

Overview

Artifact Names	Windows.Registry.Sysinternals.Eulacheck
Hunt ID	H.C2IVQGM20PMCE
Creator	sma
Creation Time	2021-05-20 06:10:10 UTC
Expiry Time	2021-05-27 06:08:15 UTC

Results

Total scheduled	
Finished clients	0
Download Results	

Available Downloads

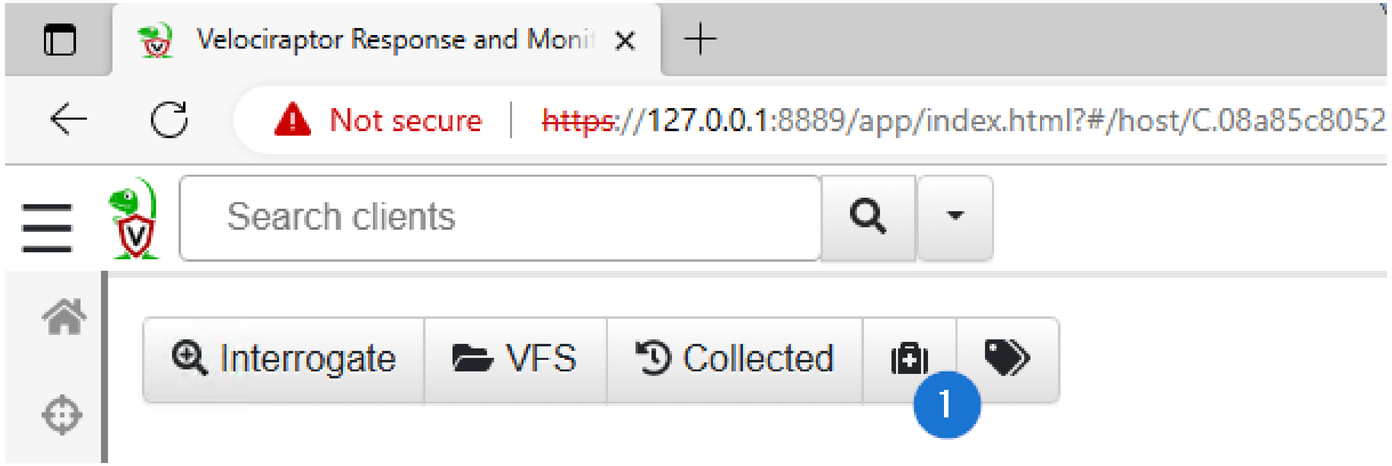
Active Containment

Gaining time

Isolate a Host

Cut Communication Instantly

Select the client and use the isolate host feature to restrict communication to server and host only.


Velociraptor Response and Monit
Not secure
~~https~~://127.0.0.1:8889/app/index.html?#/host/C.08a85c8052
Search clients
Interrogate
VFS
Collected
1

The host will be labelled «Quarantine» to be findable later on.

Collecting Files

Collecting Files

Quick Triage with Velociraptor

Incident Response may require for quick conservation of a number artifacts to be analyzed later on or being given somewhere else for analysis. With Velociraptor comes the KapeFiles Client Artifact which is a great collector of relevant files.

Approaches

- Run hunt or collection within the Velociraptor Infrastructure
- Run standalone collector with Velociraptor

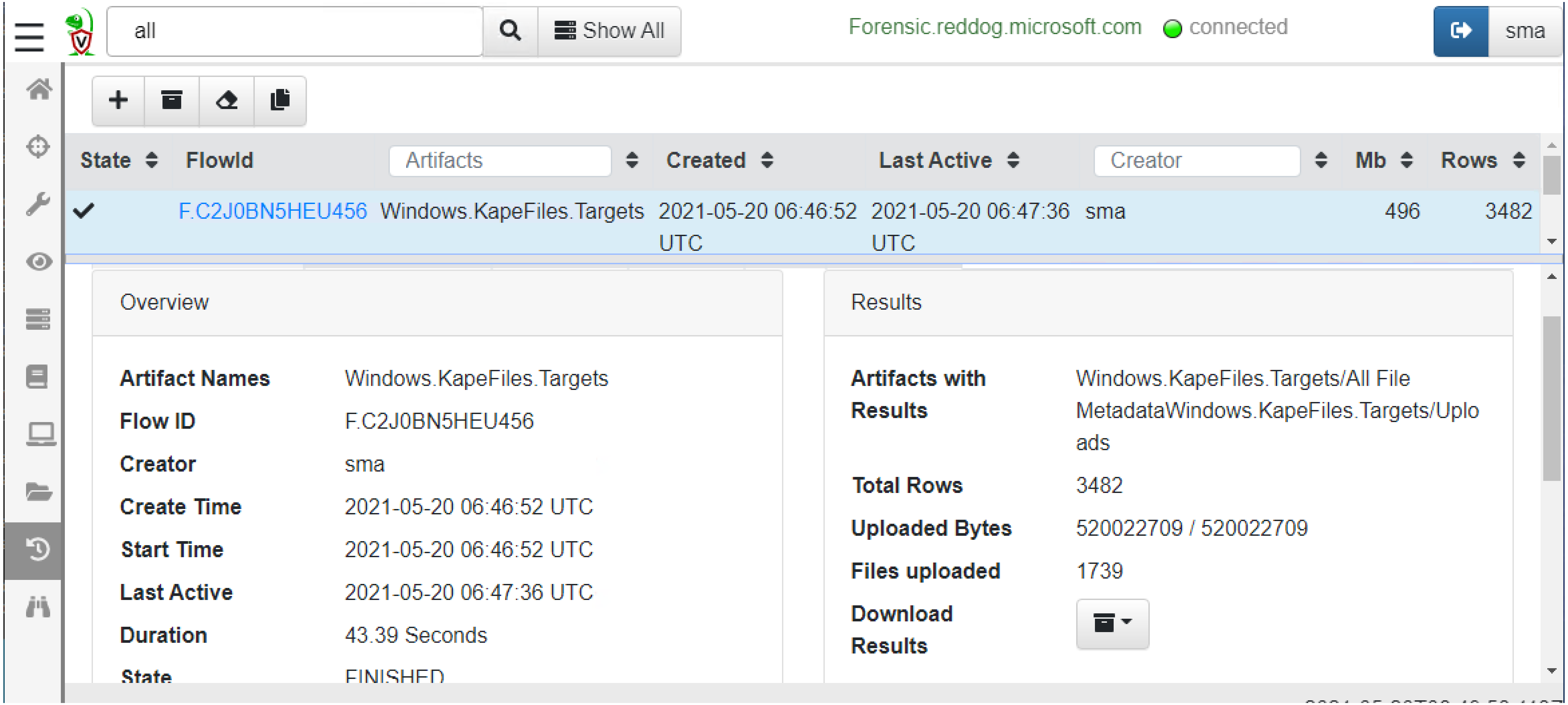
What to collect

- _BasicCollection,
- _SANS_Triage,
- _KapeTriage

… usually do well

Collecting Files

KapeFiles Collection using _SANS_Triage



State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2J0BN5HEU456	Windows.KapeFiles.Targets	2021-05-20 06:46:52 UTC	2021-05-20 06:47:36 UTC	sma	496	3482

Overview

Artifact Names	Windows.KapeFiles.Targets
Flow ID	F.C2J0BN5HEU456
Creator	sma
Create Time	2021-05-20 06:46:52 UTC
Start Time	2021-05-20 06:46:52 UTC
Last Active	2021-05-20 06:47:36 UTC
Duration	43.39 Seconds
State	FINISHED

Results

Artifacts with Results	Windows.KapeFiles.Targets/All File MetadataWindows.KapeFiles.Targets/Uploads
Total Rows	3482
Uploaded Bytes	520022709 / 520022709
Files uploaded	1739
Download Results	

Collecting Files

Potential Issues

Collection Stats from the Forensics Client

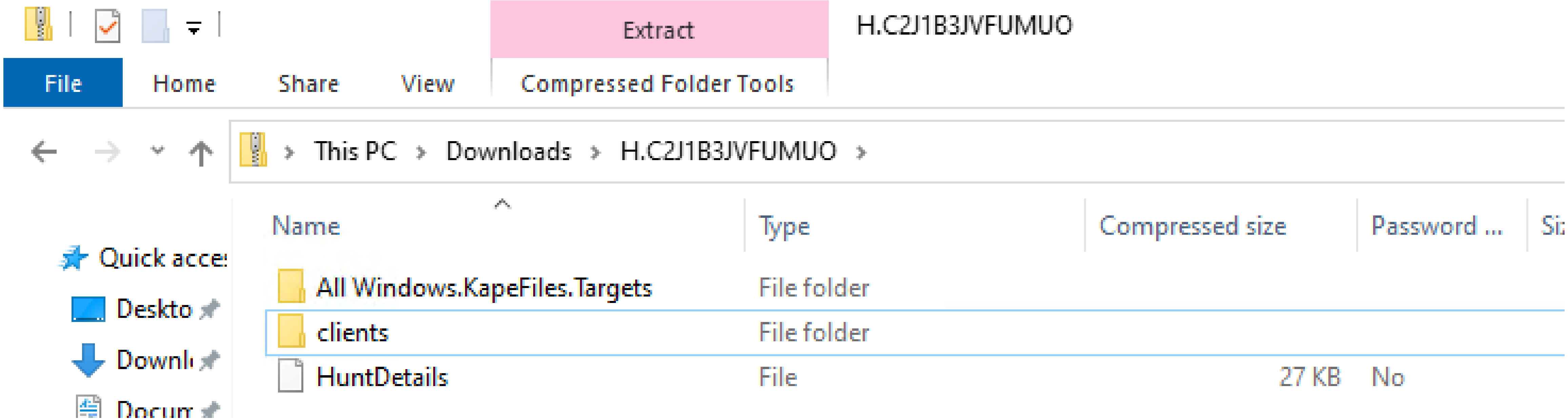
- Collection ran for 45 seconds
- Collection returned 1750 files
- 1750 files sum up to 500MB (most for the MFT, NTFS Journals, System registry)
- 1750 files zipped for download 70MB

Note, the Forensics Client is a “pretty empty” system.

Collecting large files will not result in a memory bottleneck as Velociraptor throttles clients if needed. However, extensive hunts on machines or collection of user document folders, virtual machine images, memory dumps could easily jam your server’s connection or fill your server’s disk. Be careful!

Collecting Files

Download Archives Structure, Top Level

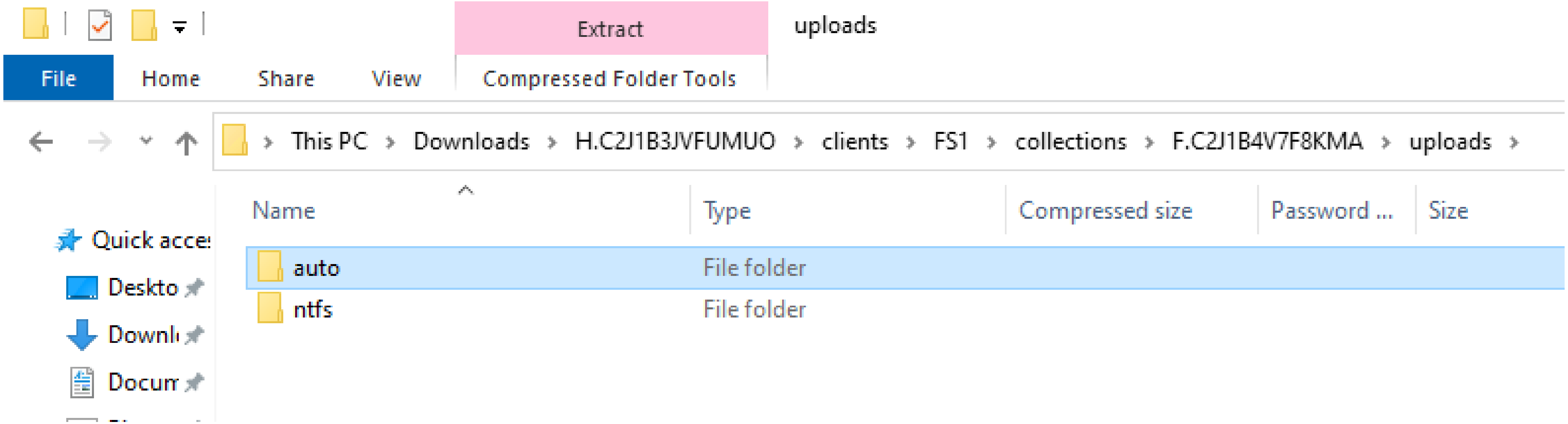


Name	Type	Compressed size	Password ...	Si:
All Windows.KapeFiles.Targets	File folder			
clients	File folder			
HuntDetails	File	27 KB	No	

- The top level includes the hunt configuration details including some meta info.
- The client folder contain the results per client workstation name.

Collecting Files

Download Archives Structure, Per Client Flow Level



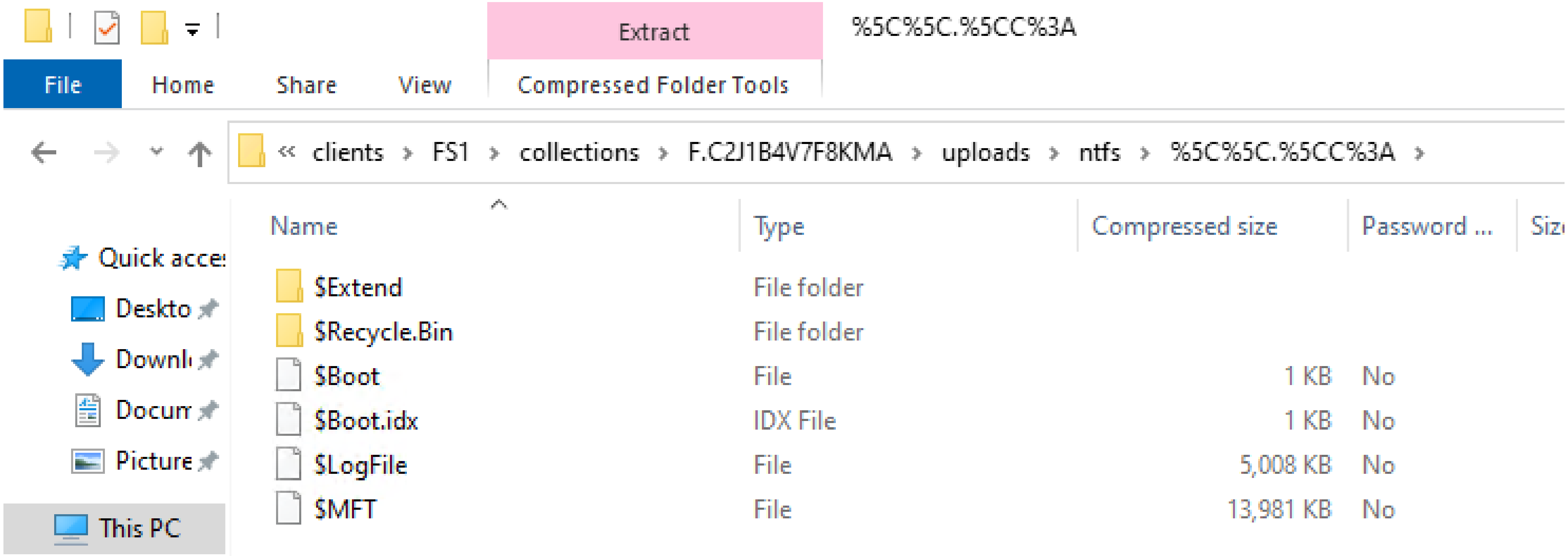
This PC › Downloads › H.C2J1B3JVFUMUO › clients › FS1 › collections › F.C2J1B4V7F8KMA › uploads ›

Name	Type	Compressed size	Password ...	Size
auto	File folder			
ntfs	File folder			

Within the collections flow folder. The uploaded files are separated by the accessor they got collected with.

Collecting Files

Download Archives Structure, Per Client Flow Level



Name	Type	Compressed size	Password ...	Siz
$Extend	File folder			
$Recycle.Bin	File folder			
$Boot	File	1 KB	No	
$Boot.idx	IDX File	1 KB	No	
$LogFile	File	5,008 KB	No	
$MFT	File	13,981 KB	No	

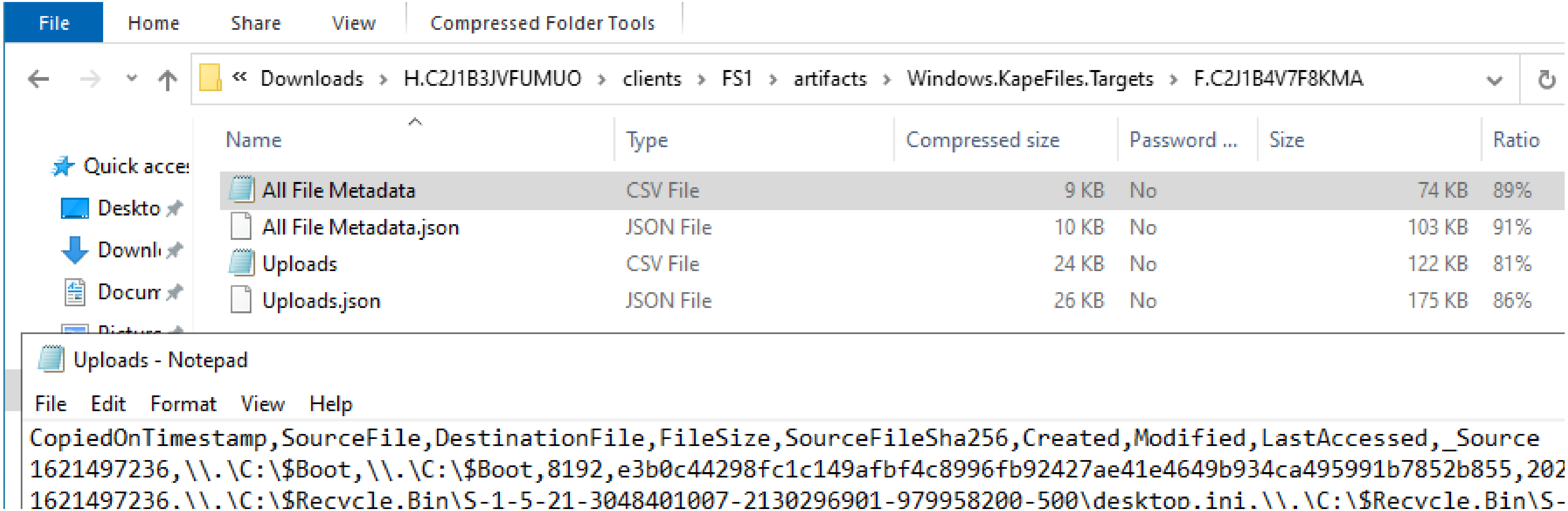
Thus $MFT et al will be available from the ntfs folder. Any common files from the auto folder.

Collecting Files

Download Archives Structure, Timestamps and Hashes

Kape does by default create a VHDX and timestomp files in it. So non-tech savvy folks may browse it like a real disk. You are better of by using the $MFT directly.

Timestamps and file hashes in the meta info files



Name	Type	Compressed size	Password ...	Size	Ratio
All File Metadata	CSV File	9 KB	No	74 KB	89%
All File Metadata.json	JSON File	10 KB	No	103 KB	91%
Uploads	CSV File	24 KB	No	122 KB	81%
Uploads.json	JSON File	26 KB	No	175 KB	86%

```
CopiedOnTimestamp,SourceFile,DestinationFile,FileSize,SourceFileSha256,Created,Modified,LastAccessed,_Source
1621497236,\\.\C:\$Boot,\\.\C:\$Boot,8192,e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855,202
1621497236,\\.\C:\$Recycle.Bin\S-1-5-21-3048401007-2130296901-979958200-500\desktop.ini,\\.\C:\$Recycle.Bin\S-
```

Manually Collecting Files

Manual Approach using the Velociraptor Binary

```
PS C:\> velociraptor.exe artifacts list *Kape*
Windows.KapeFiles.Targets
```

```
PS C:\> velociraptor.exe artifacts show Windows.KapeFiles.Targets
...
parameters:
...
  - name: _BasicCollection
    description: "Basic Collection (by Phill Moore): Thumbcache DB, at .job,
at .job, at SchedLgU.txt, at SchedLgU.txt, XML, XML, LNK Files from Recent,
...
```

```
PS C:\> velociraptor.exe artifacts collect Windows.KapeFiles.Targets
--args=_BasicCollection=Y --output=Collection_$env:computername.zip
```

Extract a Collection and Adjust (Timestomp) Files

Usefull if you want to run tools that rely on the timestamps (eg. Prefetch analysis)

Velociraptor will only record the modified time in the zip file header itself but all the times are present in the metadata file:

“Windows.KapeFiles.Targets/All File Metadata.json”

Example - command line invocation

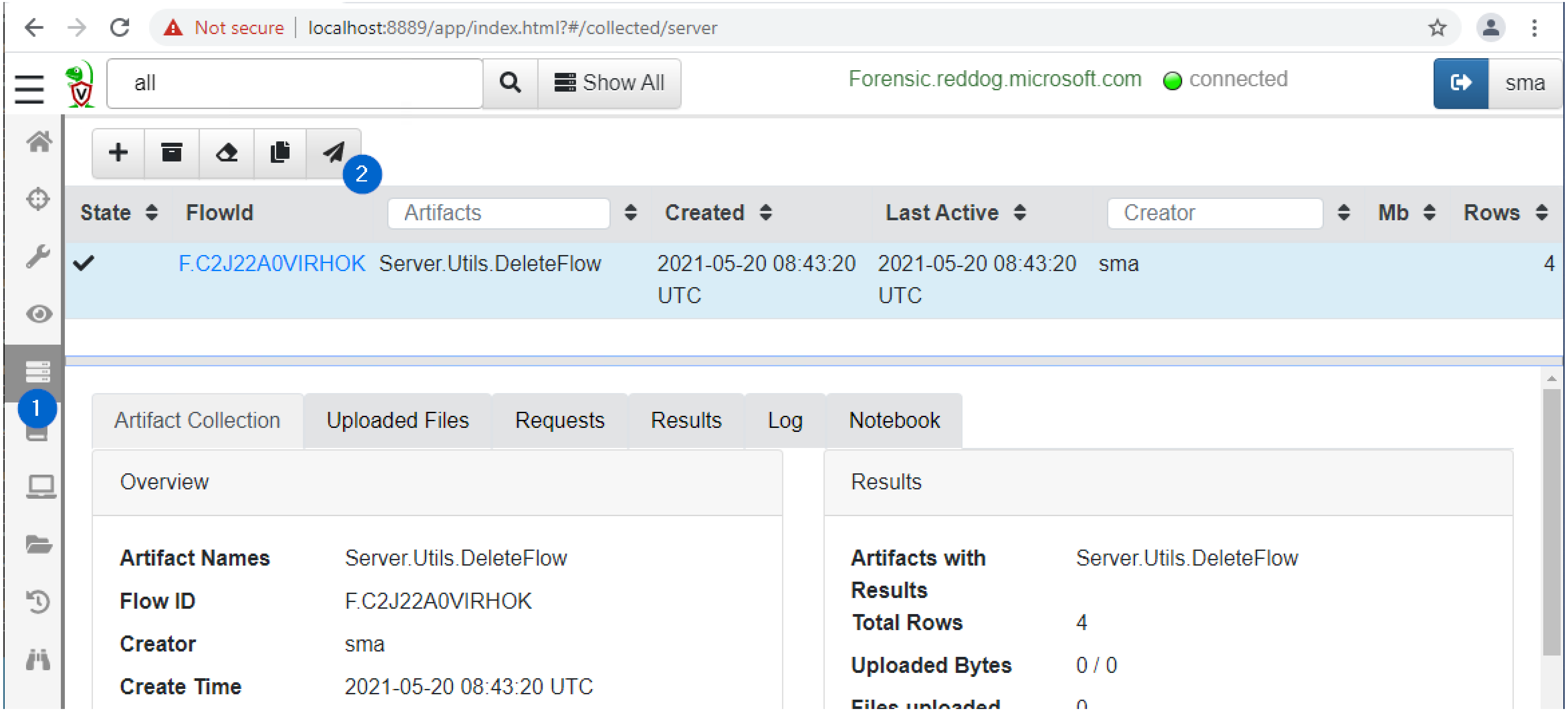
```
PS C:\> velociraptor.exe artifacts collect Windows.KapeFiles.Extract
--argsContainerPath=Collection.zip --args OutputDirectory=/tmp
```

Side note:

- Windows allows 3 timestamps to be set (MAC time except for Btime)
- Linux only allows 2 timestamps (Modified and Accessed).

Collecting Files with an Offline Collector

Open the Offline Collector Build tool, select Artifacts and Build



State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2J22A0VIRHOK	Server.Utils.DeleteFlow	2021-05-20 08:43:20 UTC	2021-05-20 08:43:20 UTC	sma		4

Overview

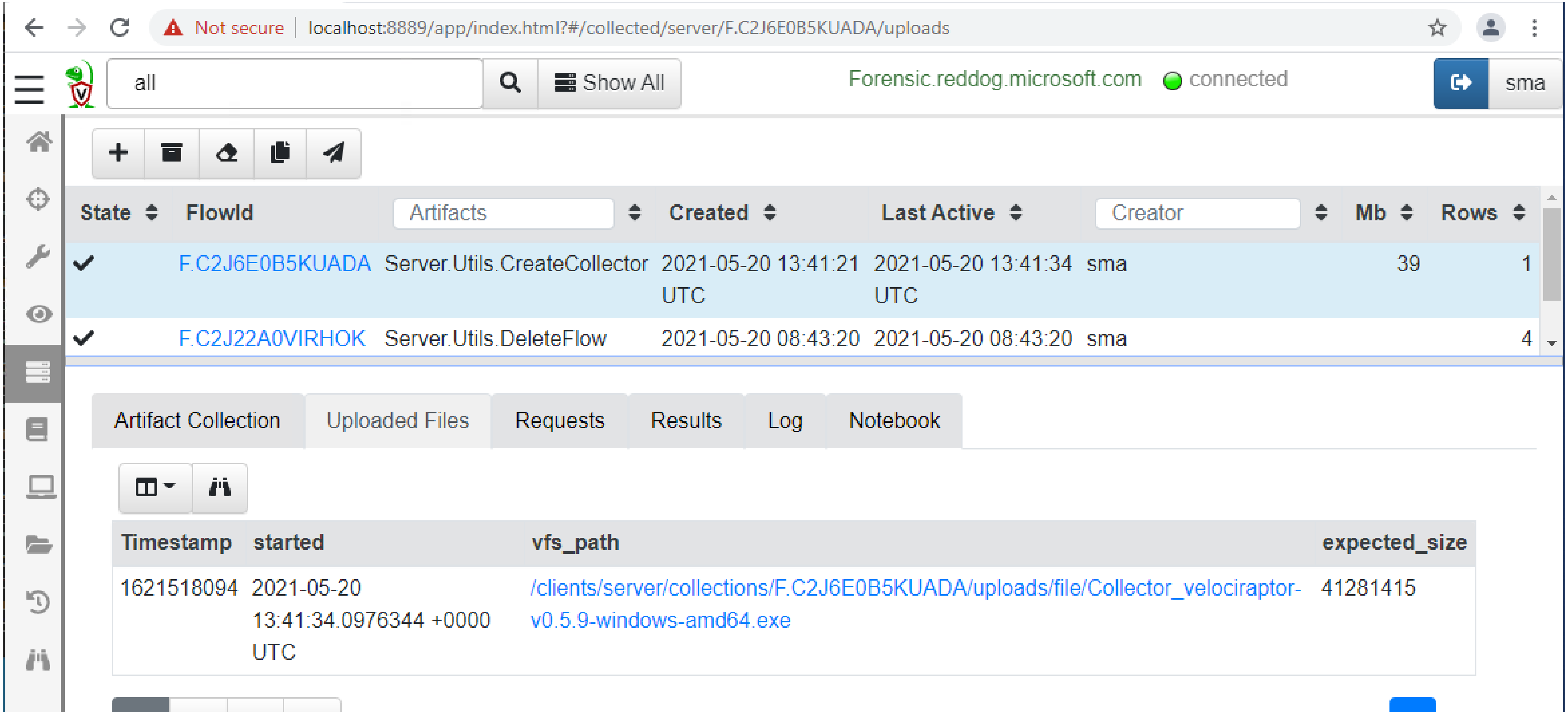
Artifact Names	Server.Utils.DeleteFlow
Flow ID	F.C2J22A0VIRHOK
Creator	sma
Create Time	2021-05-20 08:43:20 UTC

Results

Artifacts with Results	Server.Utils.DeleteFlow
Total Rows	4
Uploaded Bytes	0 / 0
Files uploaded	0

Collecting Files with an Offline Collector

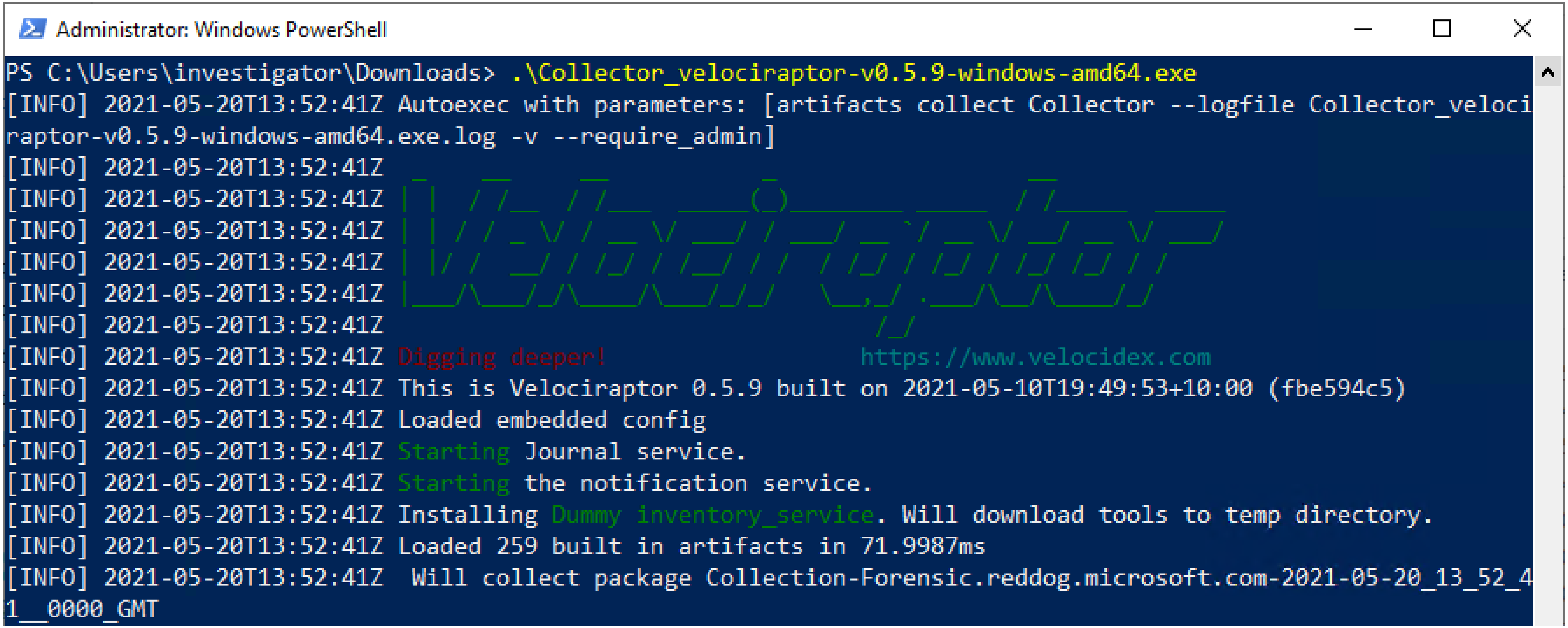
Open the Offline Collector Build tool, select Artifacts and Build


Not secure
localhost:8889/app/index.html?#/collected/server/F.C2J6E0B5KUADA/uploads
all
Show All
Forensic.reddog.microsoft.com
connected
sma
Artifact Collection
Uploaded Files
Requests
Results
Log
Notebook

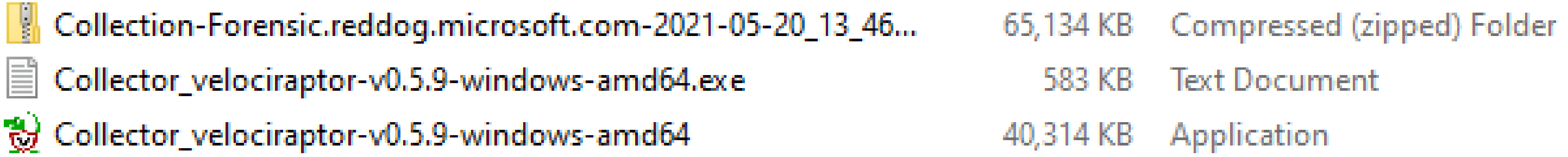
State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2J6E0B5KUADA	Server.Utils.CreateCollector	2021-05-20 13:41:21 UTC	2021-05-20 13:41:34 UTC	sma	39	1
✓	F.C2J22A0VIRHOK	Server.Utils.DeleteFlow	2021-05-20 08:43:20	2021-05-20 08:43:20	sma		4

Timestamp	started	vfs_path	expected_size
1621518094	2021-05-20 13:41:34.0976344 +0000 UTC	/clients/server/collections/F.C2J6E0B5KUADA/uploads/file/Collector_velociraptor-v0.5.9-windows-amd64.exe	41281415

Run an Offline Collector (no params needed)



```
PS C:\Users\investigator\Downloads> .\Collector_velociraptor-v0.5.9-windows-amd64.exe
[INFO] 2021-05-20T13:52:41Z Autoexec with parameters: [artifacts collect Collector --logfile Collector_veloci
raptor-v0.5.9-windows-amd64.exe.log -v --require_admin]
[INFO] 2021-05-20T13:52:41Z  _    __     __           _                 __
[INFO] 2021-05-20T13:52:41Z | |  / /__  / /___  _____(_)________ _____  / /_____  _____
[INFO] 2021-05-20T13:52:41Z | | / / _ \/ / __ \/ ___/ / ___/ __ `/ __ \/ __/ __ \/ ___/
[INFO] 2021-05-20T13:52:41Z | |/ /  __/ / /_/ / /__/ / /  / /_/ / /_/ / /_/ /_/ / /
[INFO] 2021-05-20T13:52:41Z |___/\___/_/\____/\___/_/_/   \__,_/ .___/\__/\____/_/
[INFO] 2021-05-20T13:52:41Z                                   /_/
[INFO] 2021-05-20T13:52:41Z Digging deeper!                  https://www.velocidex.com
[INFO] 2021-05-20T13:52:41Z This is Velociraptor 0.5.9 built on 2021-05-10T19:49:53+10:00 (fbe594c5)
[INFO] 2021-05-20T13:52:41Z Loaded embedded config
[INFO] 2021-05-20T13:52:41Z Starting Journal service.
[INFO] 2021-05-20T13:52:41Z Starting the notification service.
[INFO] 2021-05-20T13:52:41Z Installing Dummy inventory_service. Will download tools to temp directory.
[INFO] 2021-05-20T13:52:41Z Loaded 259 built in artifacts in 71.9987ms
[INFO] 2021-05-20T13:52:41Z  Will collect package Collection-Forensic.reddog.microsoft.com-2021-05-20_13_52_4
1__0000_GMT
```



Collection-Forensic.reddog.microsoft.com-2021-05-20_13_46...	65,134 KB	Compressed (zipped) Folder
Collector_velociraptor-v0.5.9-windows-amd64.exe	583 KB	Text Document
Collector_velociraptor-v0.5.9-windows-amd64	40,314 KB	Application

SPEEEEEEEEEED!

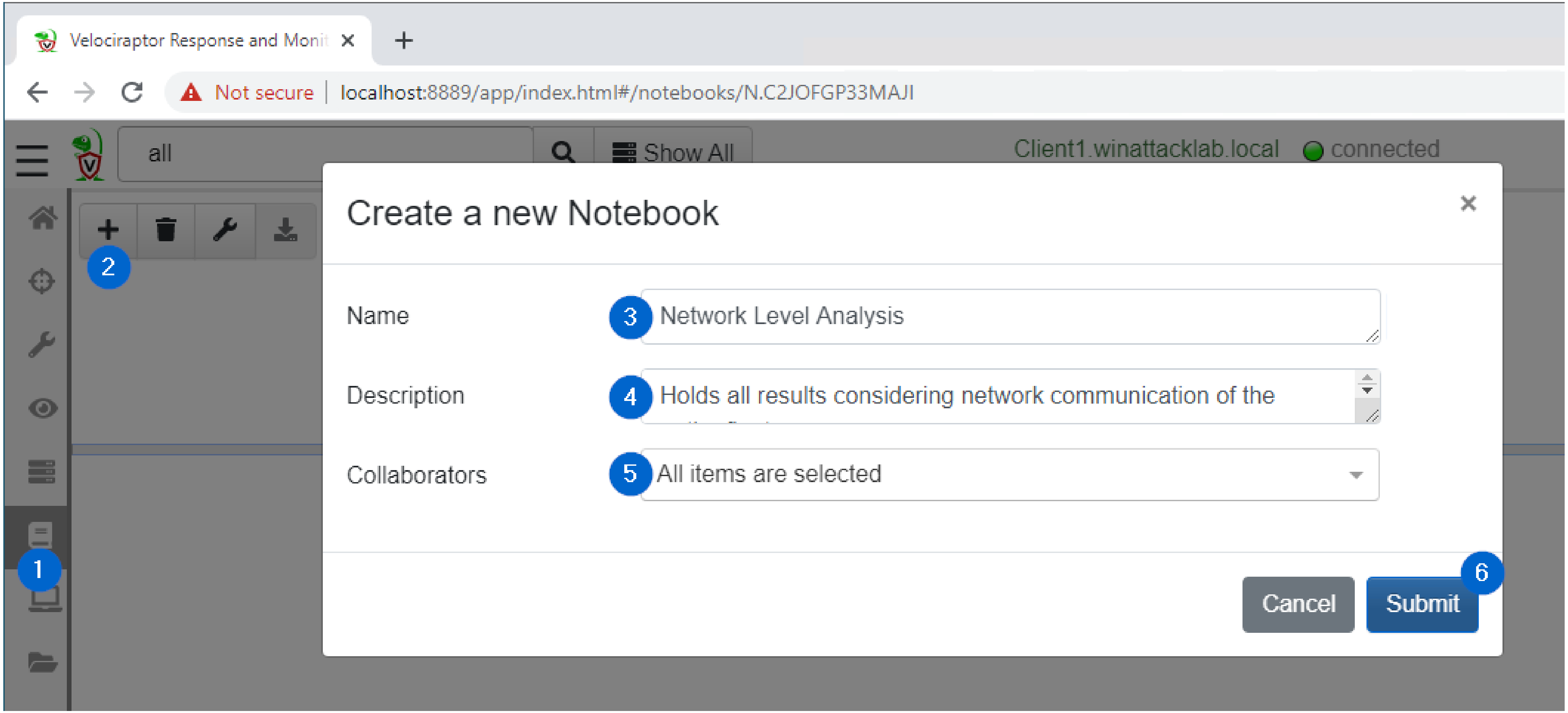
Velociraptor Speed to Numberz

Action, Artifact	Time
Hunt for the full file path on a target systems using Windows.System.CmdShell with param “cmd.exe /c dir C:\Users\mpotter\compass-test-file.txt”	Instant
Hunt for filename only using Windows.Forensics.FilenameSearch which searches the $MFT	30 sec
Creating a hash DB on target by Generic.Forensic.LocalHashes.Glob filtered for C:\Users\*** (4400 files).	4 min
Use Generic.Forensic.LocalHashes.Query to **search a hash DB**	Instant
Hunt for successful logons using Windows.EventLogs.ExplicitLogon **on Client**	10 sec
Hunt for successful logons using Windows.EventLogs.ExplicitLogon **on DC**	50 sec
Yara scan processes on a Windows box for a simple string	5 min
Collecting 4GB of physical memory on a client => 1.2GB Zip Archive	12 min

Notebooks

Velociraptor Notebooks

Create a New Notebook



Velociraptor Response and Monit
Not secure
localhost:8889/app/index.html#/notebooks/N.C2JOFGP33MAJI
all
Show All
Client1.winattacklab.local
connected
1
2
Cancel
Submit
6

Create a new Notebook

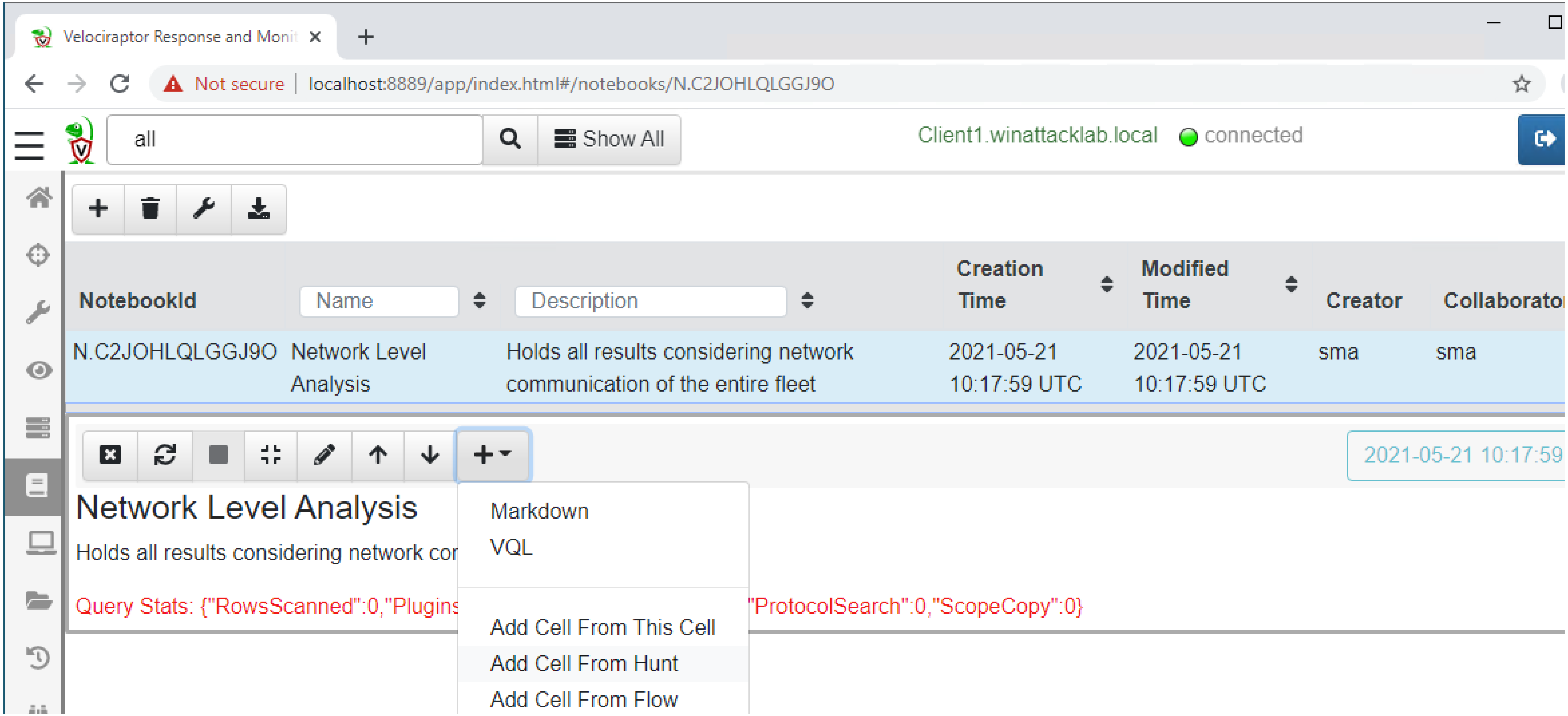
3 Name: Network Level Analysis

4 Description: Holds all results considering network communication of the

5 Collaborators: All items are selected

Velociraptor Notebooks

Edit Cells or Add New Cells using Existing Results (Hunts and Flows)



NotebookId	Name	Description	Creation Time	Modified Time	Creator	Collaborato
N.C2JOHLQLGGJ9O	Network Level Analysis	Holds all results considering network communication of the entire fleet	2021-05-21 10:17:59 UTC	2021-05-21 10:17:59 UTC	sma	sma

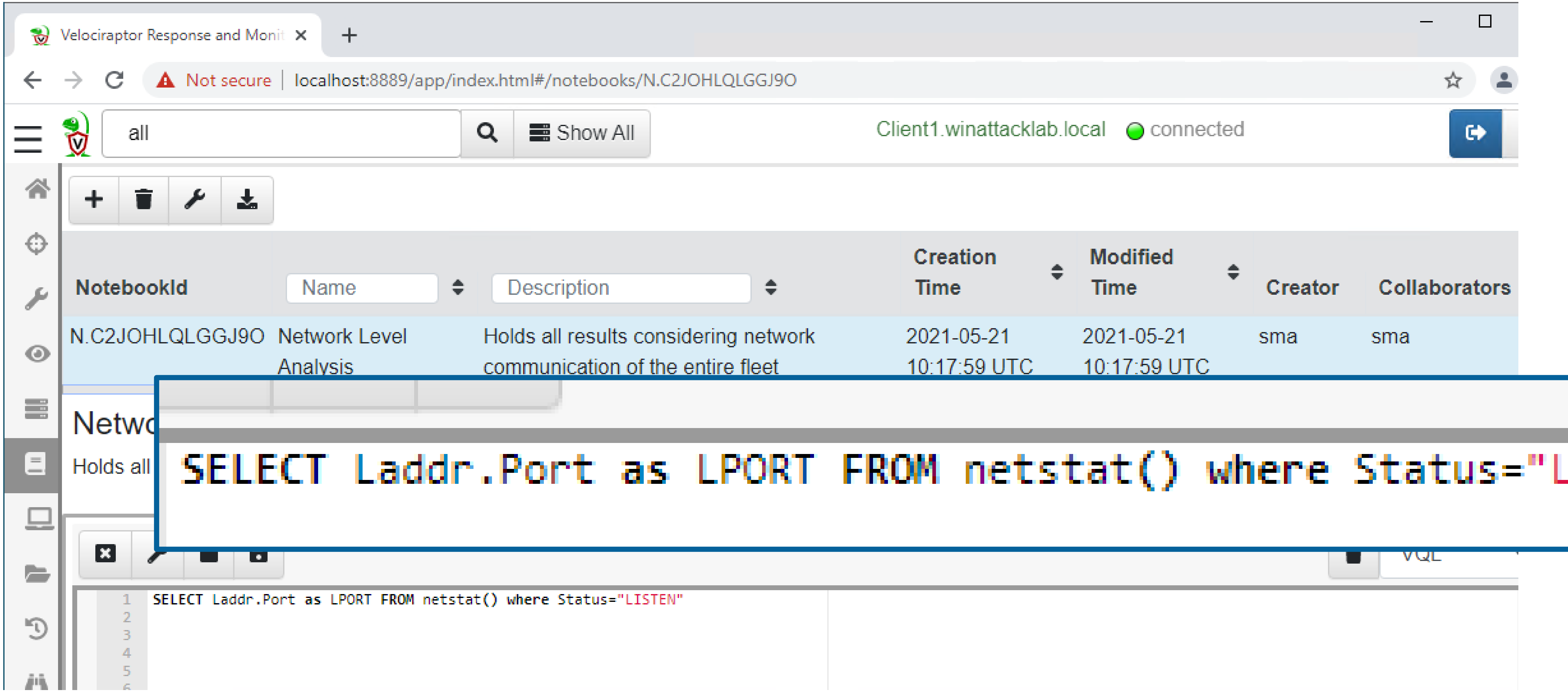
Network Level Analysis

Holds all results considering network cor

Query Stats: {"RowsScanned":0,"Plugins "ProtocolSearch":0,"ScopeCopy":0}

Velociraptor Notebooks

Add New VQL Query



Velociraptor Response and Monit
Not secure
localhost:8889/app/index.html#/notebooks/N.C2JOHLQLGGJ9O
all
Show All
Client1.winattacklab.local
connected
Netwo
Holds all
VQL

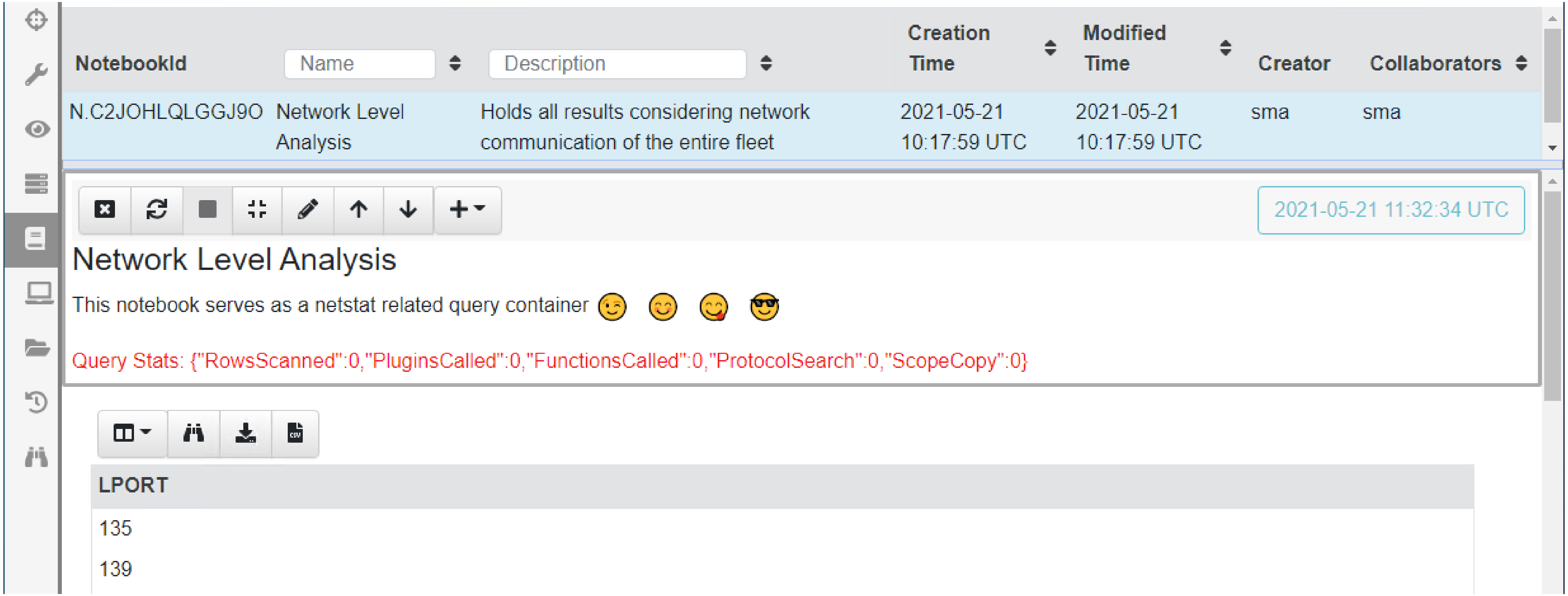
NotebookId	Name	Description	Creation Time	Modified Time	Creator	Collaborators
N.C2JOHLQLGGJ9O	Network Level Analysis	Holds all results considering network communication of the entire fleet	2021-05-21 10:17:59 UTC	2021-05-21 10:17:59 UTC	sma	sma

SELECT Laddr.Port as LPORT FROM netstat() where Status="L

```
SELECT Laddr.Port as LPORT FROM netstat() where Status="LISTEN"
```

Velociraptor Notebooks

Results in Notebook Cells



NotebookId	Name	Description	Creation Time	Modified Time	Creator	Collaborators
N.C2JOHLQLGGJ9O	Network Level Analysis	Holds all results considering network communication of the entire fleet	2021-05-21 10:17:59 UTC	2021-05-21 10:17:59 UTC	sma	sma

2021-05-21 11:32:34 UTC

Network Level Analysis

This notebook serves as a netstat related query container 😉 😊 😋 😎

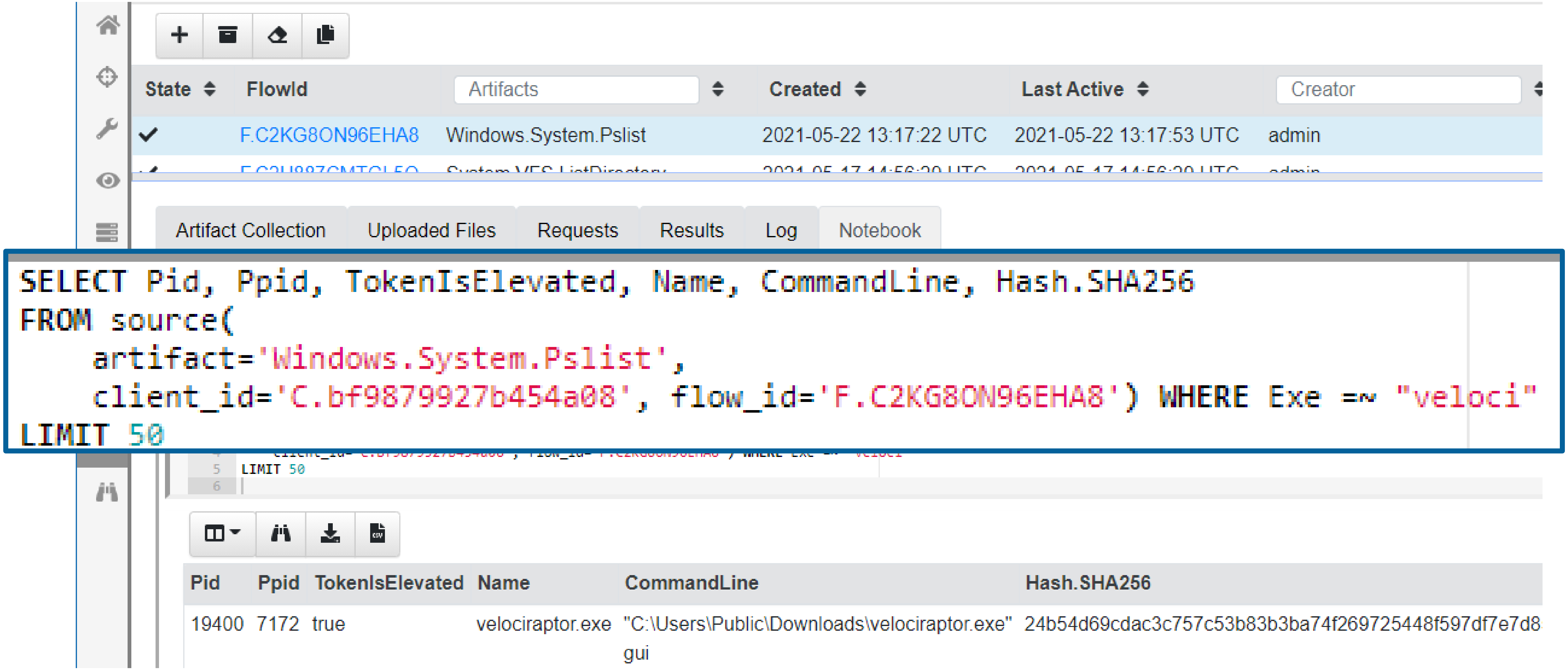
Query Stats: {"RowsScanned":0,"PluginsCalled":0,"FunctionsCalled":0,"ProtocolSearch":0,"ScopeCopy":0}

LPORT
135
139

Results or errors are shown as soon as the VQL cell is saved.

Velociraptor Notebooks

Tailor Flow Results (e.g. filter a Pslist Artifact output)



State	FlowId	Artifacts	Created	Last Active	Creator
✓	F.C2KG8ON96EHA8	Windows.System.Pslist	2021-05-22 13:17:22 UTC	2021-05-22 13:17:53 UTC	admin
✓	[illegible]	System.VFS.ListDirectory	2021-05-17 14:56:20 UTC	2021-05-17 14:56:20 UTC	admin

```
SELECT Pid, Ppid, TokenIsElevated, Name, CommandLine, Hash.SHA256
FROM source(
    artifact='Windows.System.Pslist',
    client_id='C.bf9879927b454a08', flow_id='F.C2KG8ON96EHA8') WHERE Exe =~ "veloci"
LIMIT 50
```

Pid	Ppid	TokenIsElevated	Name	CommandLine	Hash.SHA256
19400	7172	true	velociraptor.exe	"C:\Users\Public\Downloads\velociraptor.exe" gui	24b54d69cdac3c757c53b83b3ba74f269725448f597df7e7d8

Velociraptor Query Language

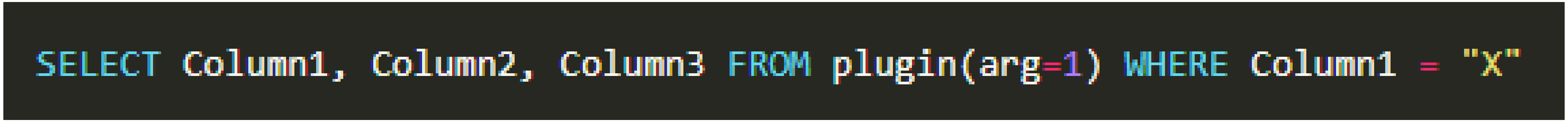
Querying the hell out of your infrastructure

Velociraptor Query Language

Introduction to VQL

What is VQL

- It looks like SQL
- Everything in Velociraptor is basically returning Table (Resultset)
- Functions and plugins are the major accelerators



```
SELECT Column1, Column2, Column3 FROM plugin(arg=1) WHERE Column1 = "X"
```

Column Specification

Plugin Clause

Filter Clause

Reference https://www.velocidex.com/docs/vql_reference/

Velociraptor Query Language

Helpful Syntax

Commenting your queries works with

`-- comment` or `// comment`

If you want to match strings

```
SELECT * FROM pslist() WHERE Exe =~ "veloci"
```

Put values or entire queries into a variable using LET

```
LET test = "gugus"
LET test = SELECT * FROM pslist()  // references the query (~pointer)
LET test <= SELECT * FROM pslist() // fills test with the result
```

Limit damage ;) using LIMIT

```
SELECT * FROM glob(globs="C:/**") LIMIT 5
```

It is suggested to use / for paths, always. \\ might work for Windows clients, though

Velociraptor Query Language

Logs, List of Files or Registry Entries

Use log() to do Candle-Light Debugging

```
SELECT Null FROM pslist() WHERE log(message="yeah, hit line")
```

Use the glob plugin to get a list of files easily.

```
SELECT Name FROM glob(globs="C:/Users/**/Downloads/*.ex?") LIMIT 5
```

Globs support wildcards such as

```
?    for a single letter
*    part of a string
**   used to traverse recursively into folder
```

Globs support different file accessors e.g. the Registry

```
SELECT FullPath, Name, Data.type, Data.value FROM
glob(globs="HKEY_USERS/*/Software/**/*", accessor="reg")
```

Reference https://www.velocidex.com/docs/vql_reference/

Velociraptor Query Language

Branching and Looping

Use if to branch depending on a condition

```
SELECT * FROM if(
    condition=Exe =~ "chrome",
    then={ expression or query },
    else={ expression or query }    // else is optional
)
```

You may loop over result sets using foreach applying a sub query

```
LET foo = SELECT * FROM pslist() where Exe =~ "chrome" LIMIT 5
SELECT * FROM foreach(
    row=foo,
    query={SELECT * FROM handles(pid=Pid)}
)
```

Reference https://www.velocidex.com/docs/vql_reference/

Velociraptor Query Language Exercise

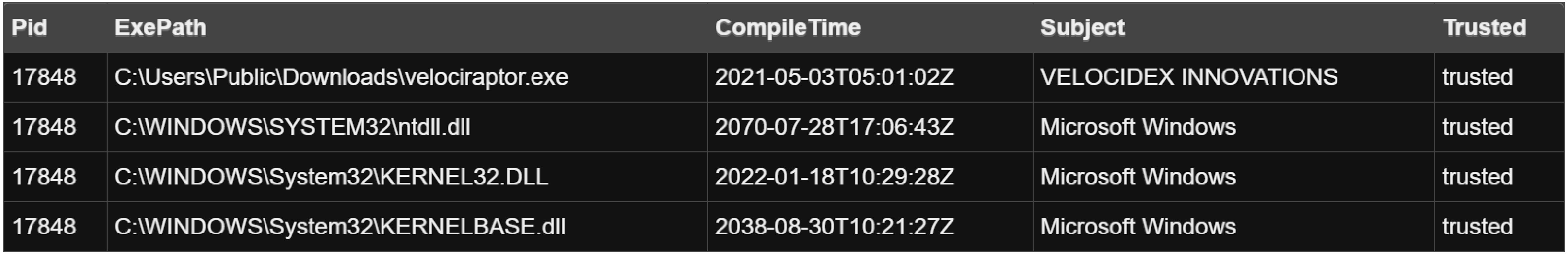
Let’s give it a try

- Switch to the notebook view
- Create a new notebook
- Enter some text and copy&paste a screenshot
- Create a new VQL cell
- **Create a query that lists loaded DLLs for Velociraptor including compile time and signature**
  - Find the Velociraptor Pid (pslist)
  - Create a list of DLLs (modules, Pid)
  - Get the compile time (parse_pe, ExePath)
  - List if the binary is trusted and the signing subject (authenticode, Exe)

Velociraptor Query Language Exercise Solution

Create a query that lists loaded DLLs for Velociraptor including compile time and signature

```
LET pids = SELECT * FROM pslist() WHERE Exe =~ "veloci"
SELECT * FROM foreach(
   row = pids,
   query = {
      SELECT Pid, ExePath, parse_pe(file=ExePath).FileHeader.TimeDateStamp as
      CompileTime, authenticode(filename=ExePath).SubjectName as Subject,
      authenticode(filename=ExePath).Trusted as Trusted FROM modules(pid=Pid)
   })
```



Pid	ExePath	CompileTime	Subject	Trusted
17848	C:\Users\Public\Downloads\velociraptor.exe	2021-05-03T05:01:02Z	VELOCIDEX INNOVATIONS	trusted
17848	C:\WINDOWS\SYSTEM32\ntdll.dll	2070-07-28T17:06:43Z	Microsoft Windows	trusted
17848	C:\WINDOWS\System32\KERNEL32.DLL	2022-01-18T10:29:28Z	Microsoft Windows	trusted
17848	C:\WINDOWS\System32\KERNELBASE.dll	2038-08-30T10:21:27Z	Microsoft Windows	trusted

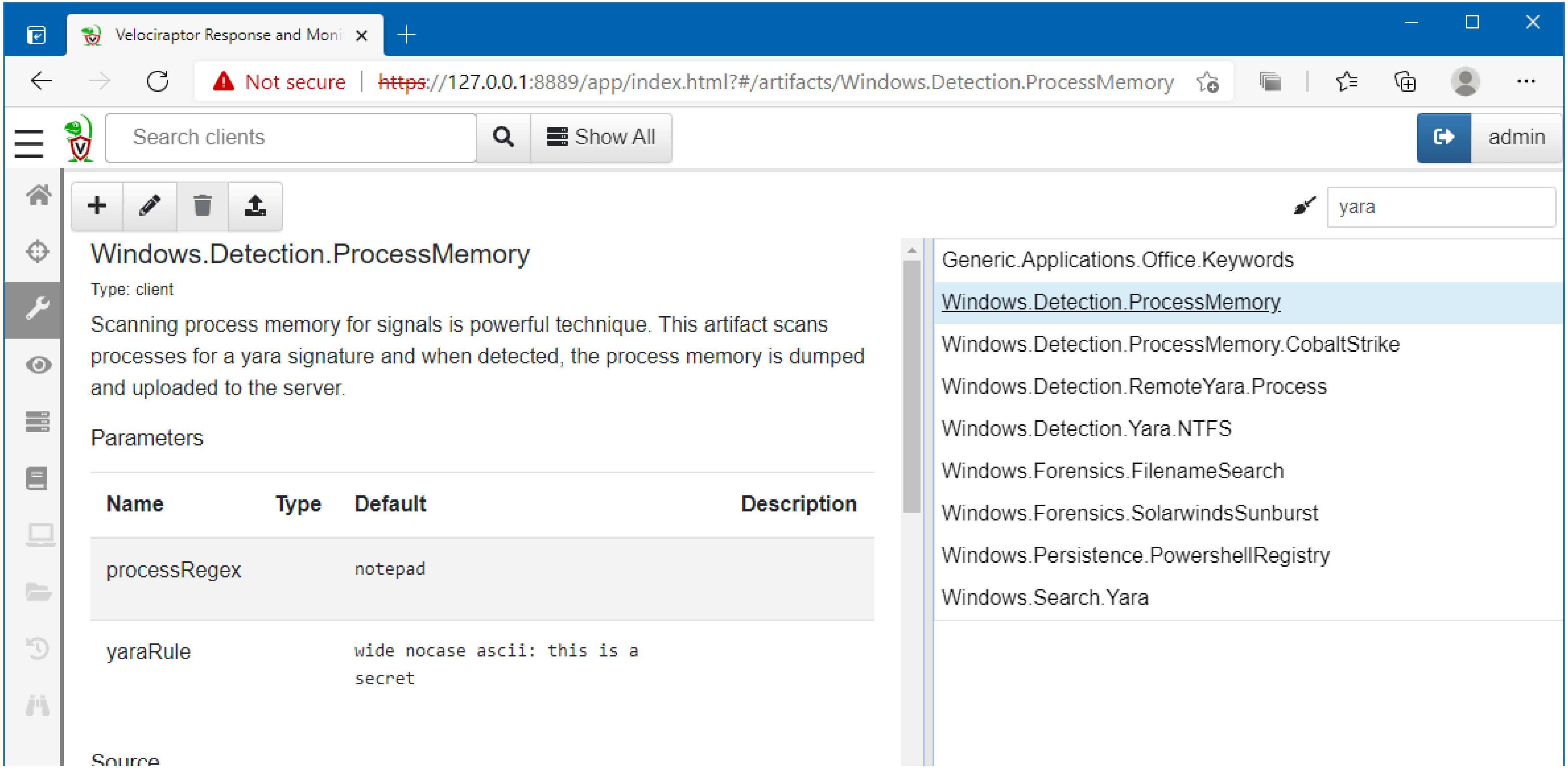
YARA Hunting

YARA Recap

```
rule silent_banker : banker
{
    meta:
        description = "This is just an example"
        threat_level = 3
        in_the_wild = true

    strings:
        $a = {6A 40 68 00 30 00 00 6A 14 8D 91}
        $b = {8D 4D B0 2B C1 83 C0 27 99 6A 4E 59 F7 F9}
        $c = "UVODFRYSIHLNWPEJXQZAKCBGMT"

    condition:
        $a or $b or $c
}
```

http://virustotal.github.io/yara/

YARA in Velociraptor


Velociraptor Response and Moni
Not secure
https://127.0.0.1:8889/app/index.html?#/artifacts/Windows.Detection.ProcessMemory
Search clients
Show All
admin
yara

Windows.Detection.ProcessMemory

Type: client

Scanning process memory for signals is powerful technique. This artifact scans processes for a yara signature and when detected, the process memory is dumped and uploaded to the server.

Parameters

Name	Type	Default	Description
processRegex		`notepad`	
yaraRule		`wide nocase ascii: this is a secret`	

Source

- Generic.Applications.Office.Keywords
- Windows.Detection.ProcessMemory
- Windows.Detection.ProcessMemory.CobaltStrike
- Windows.Detection.RemoteYara.Process
- Windows.Detection.Yara.NTFS
- Windows.Forensics.FilenameSearch
- Windows.Forensics.SolarwindsSunburst
- Windows.Persistence.PowershellRegistry
- Windows.Search.Yara

YARA Performance

You can get yara rules from many sources (threat intel, blog posts etc)

YARA is really a **first level triage tool**:

- Depending on signature many false positives expected
- Some signatures are extremely specific so make a great signal

Speed things up

- Try to collect additional context around the hits to eliminate false positives.

Avoid DoS

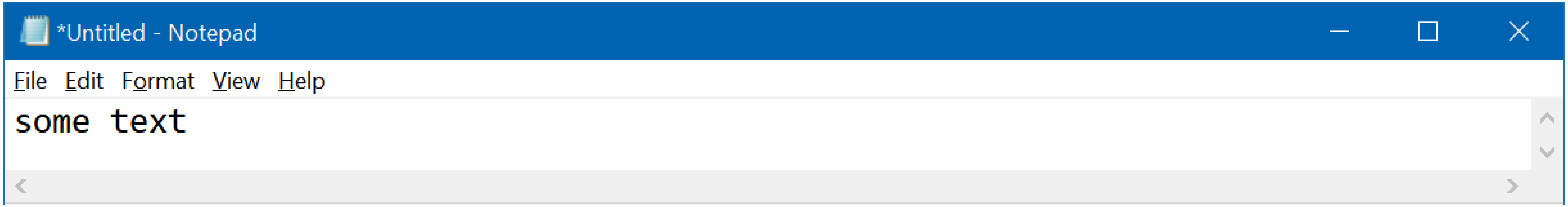
- Yara scanning is relatively expensive! Consider more targeted glob expressions and client-side throttling since usually YARA scanning is not time critical.

Source: Velocidex Enterprise

Velociraptor YARA (inline) Hunt Example

Let’s try to find something memory resident. Thus, you need to prepare a system with a specific string in memory.

Fire-up a Notepad* and enter some text. Well, it could be something else of course.



some text

Try to find the “malicious” process and list its ID, name and executable path

- Create a yara rule that hits the “specific string” or “some text” (string, and condition)

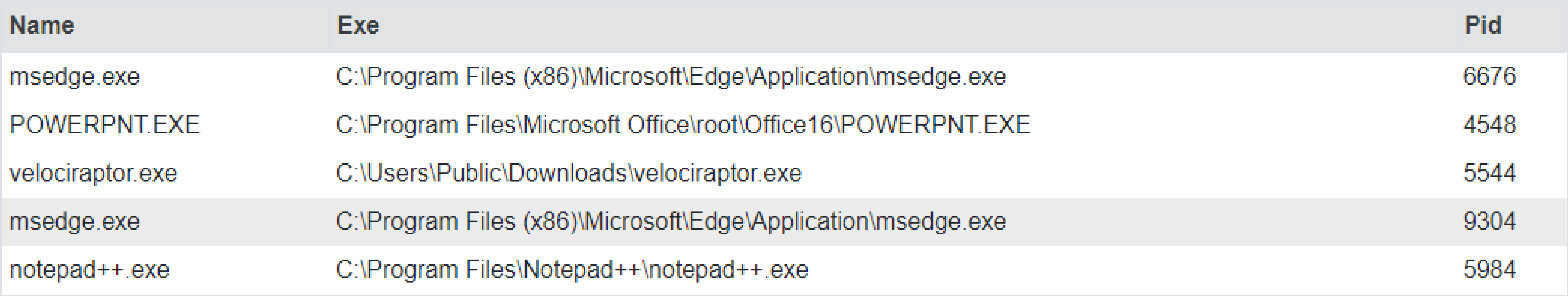
```
LET r001 = 'rule detector { strings: $srch = "some text" condition : $srch }'
```

- Enumerate all processes (pslist)
- For every process, do a yara search (proc_yara, Pid)

Velociraptor YARA (inline) Hunt Example Solution

Finding a process is no magic.

```
LET rule = 'rule dtect { strings: $srch = "Secret_Name" condition : $srch }'
LET pid = SELECT Pid, Exe, Name FROM pslist()
LET qry = SELECT Name, Exe, Pid from proc_yara( pid=Pid, rules=rule)

SELECT * FROM foreach(row=pid, query=qry)
```

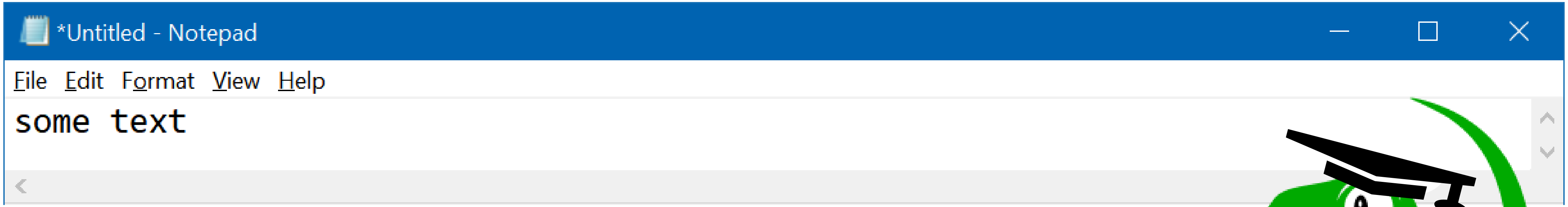


Name	Exe	Pid
msedge.exe	C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe	6676
POWERPNT.EXE	C:\Program Files\Microsoft Office\root\Office16\POWERPNT.EXE	4548
velociraptor.exe	C:\Users\Public\Downloads\velociraptor.exe	5544
msedge.exe	C:\Program Files (x86)\Microsoft\Edge\Application\msedge.exe	9304
notepad++.exe	C:\Program Files\Notepad++\notepad++.exe	5984

Velociraptor YARA (file) Hunt Example

Let’s try to find something based on a rule file. We need something to be found and thus, you need to prepare a file with a specific string in it.

Fire-up a Notepad* and enter some text. Save it somewhere but use the ending “.txt”



some text

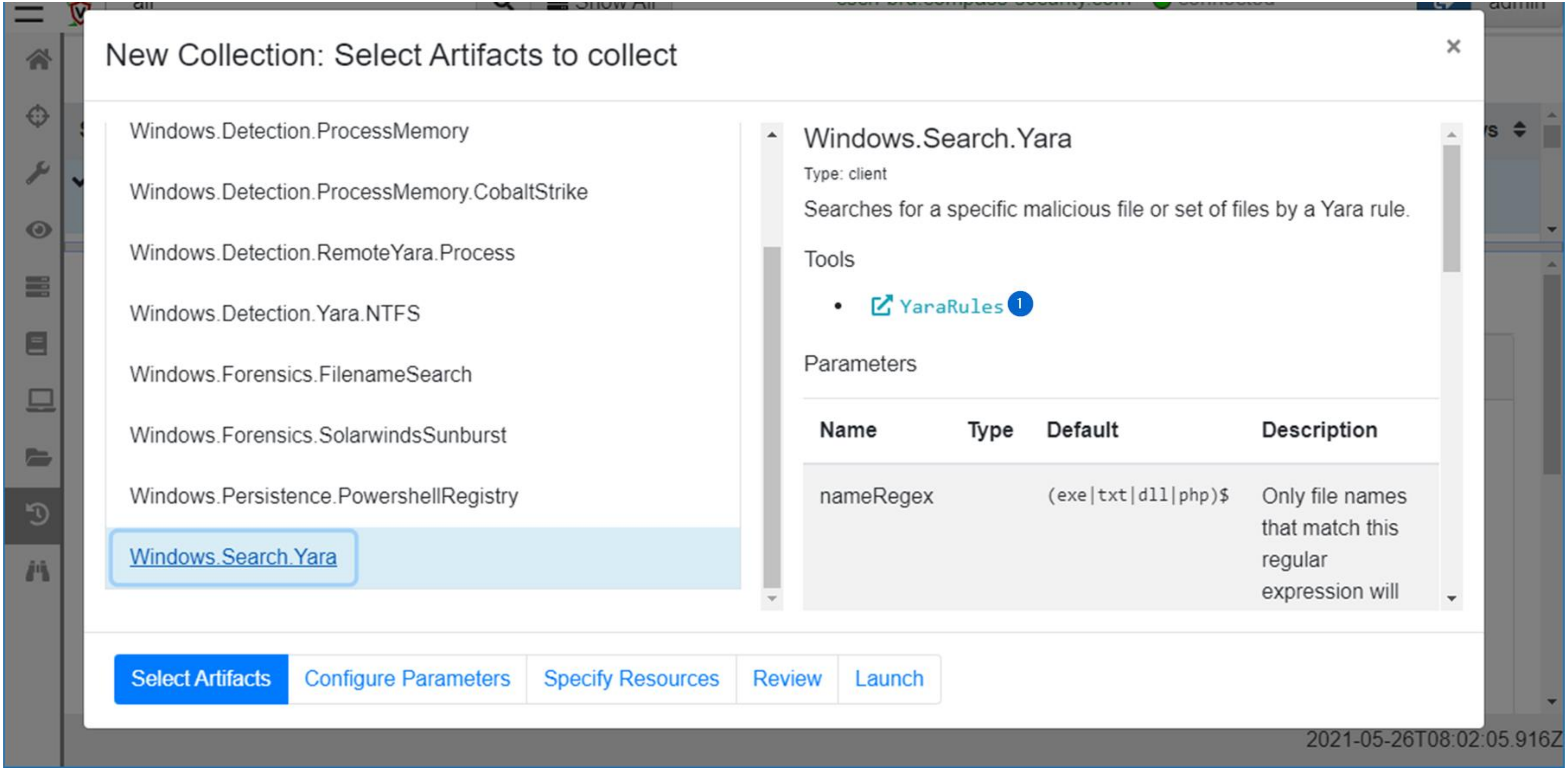
Try to find the “malicious” file using the Windows.Search.YARA Artifact.

- Save a yara rule into a file (string, and condition)

```
rule dtect { strings: $srch = "some text" condition : $srch }
```

- Upload the file using the Windows.Search.YARA Upload button
- Run the flow or hunt (you maybe want to narrow the scope to *.txt)

Velociraptor YARA (file) Hunt Example Solution



New Collection: Select Artifacts to collect

- Windows.Detection.ProcessMemory
- Windows.Detection.ProcessMemory.CobaltStrike
- Windows.Detection.RemoteYara.Process
- Windows.Detection.Yara.NTFS
- Windows.Forensics.FilenameSearch
- Windows.Forensics.SolarwindsSunburst
- Windows.Persistence.PowershellRegistry
- Windows.Search.Yara

Windows.Search.Yara

Type: client

Searches for a specific malicious file or set of files by a Yara rule.

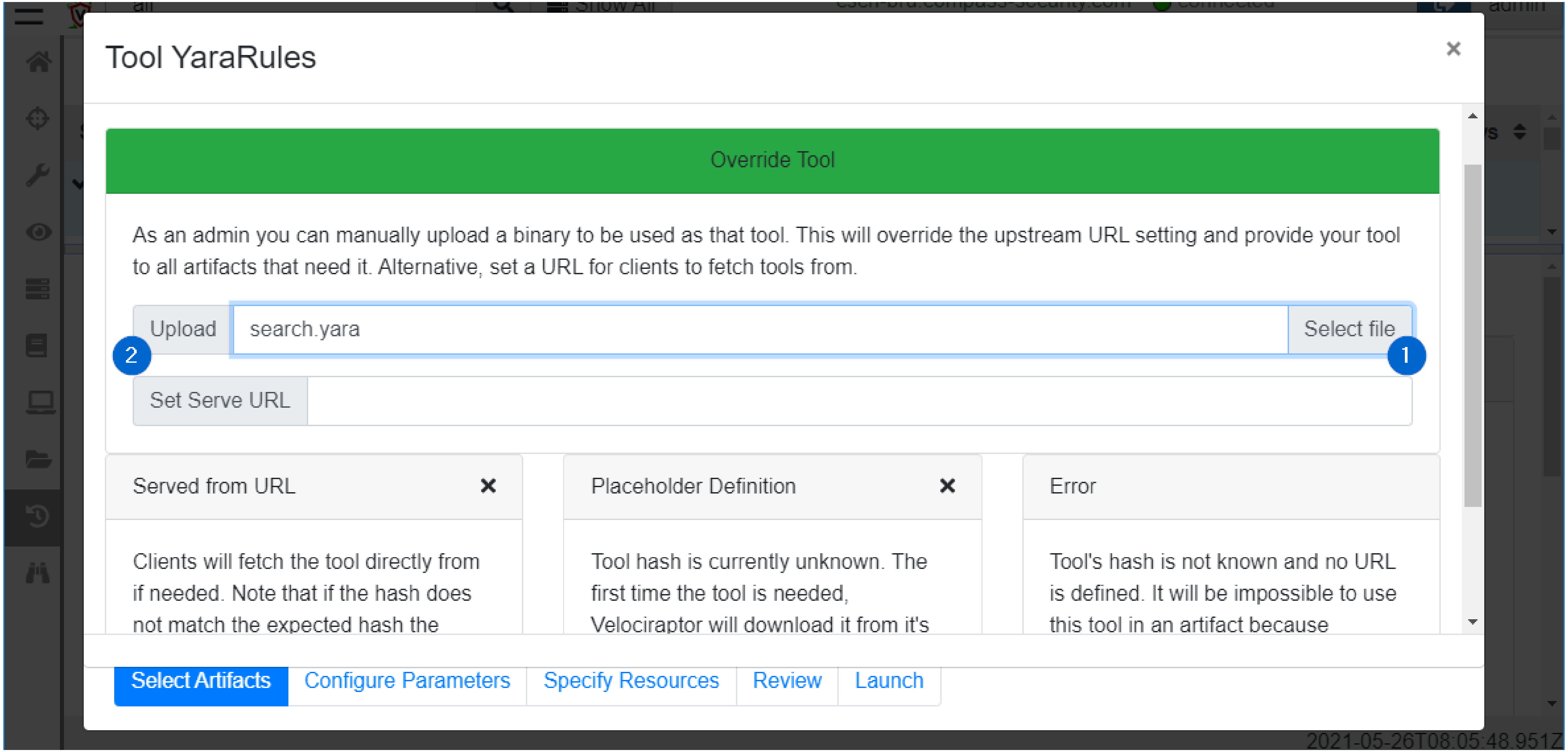
Tools

- YaraRules 1

Parameters

Name	Type	Default	Description
nameRegex		(exe|txt|dll|php)$	Only file names that match this regular expression will

Velociraptor YARA (file) Hunt Example Solution



Tool YaraRules

Override Tool

As an admin you can manually upload a binary to be used as that tool. This will override the upstream URL setting and provide your tool to all artifacts that need it. Alternative, set a URL for clients to fetch tools from.

Upload: search.yara — Select file

Set Serve URL

Served from URL

Clients will fetch the tool directly from if needed. Note that if the hash does not match the expected hash the

Placeholder Definition

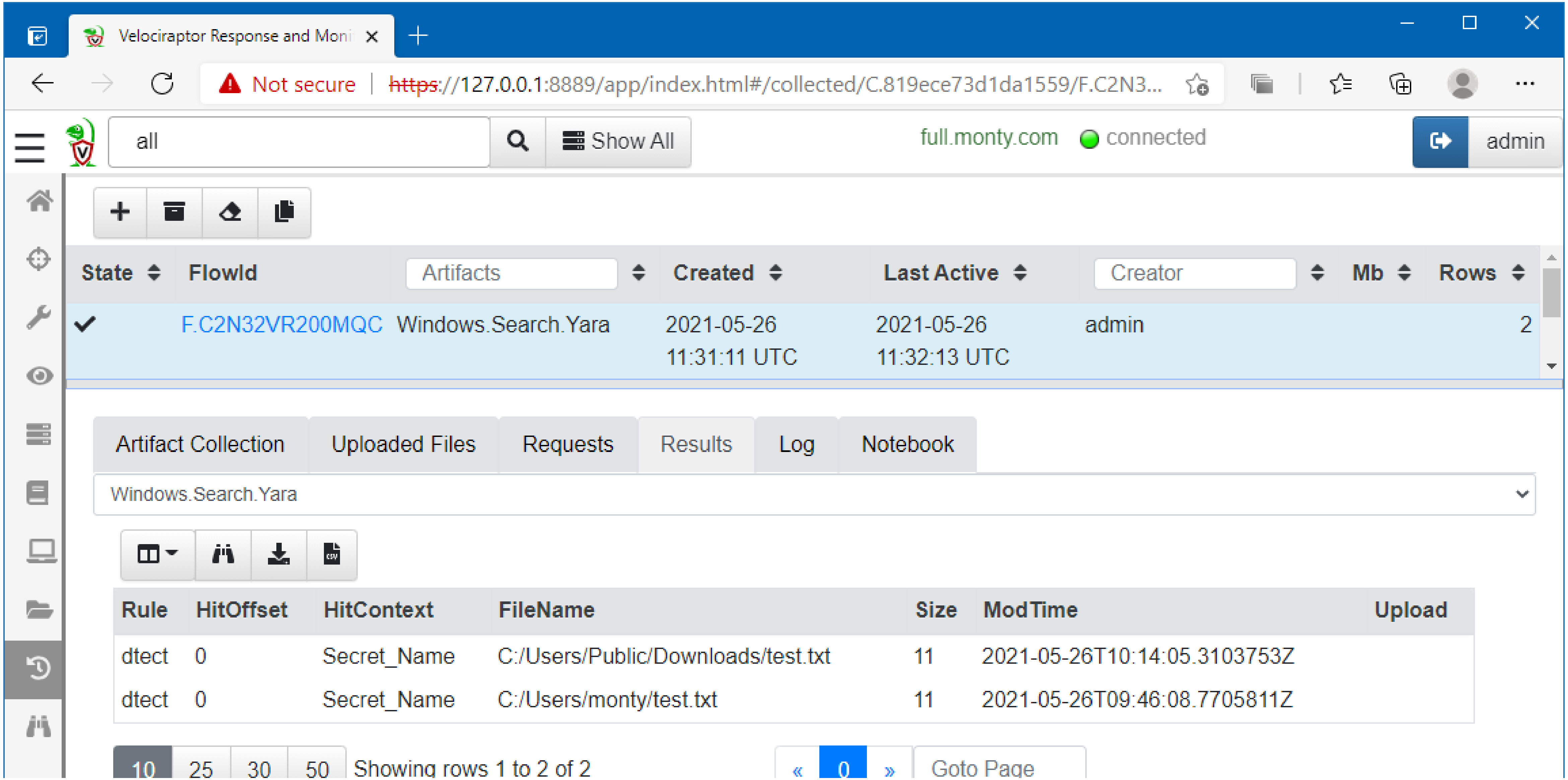
Tool hash is currently unknown. The first time the tool is needed, Velociraptor will download it from it's

Error

Tool's hash is not known and no URL is defined. It will be impossible to use this tool in an artifact because

2021-05-26T08:05:48.951Z

Velociraptor YARA (file) Hunt Example Solution



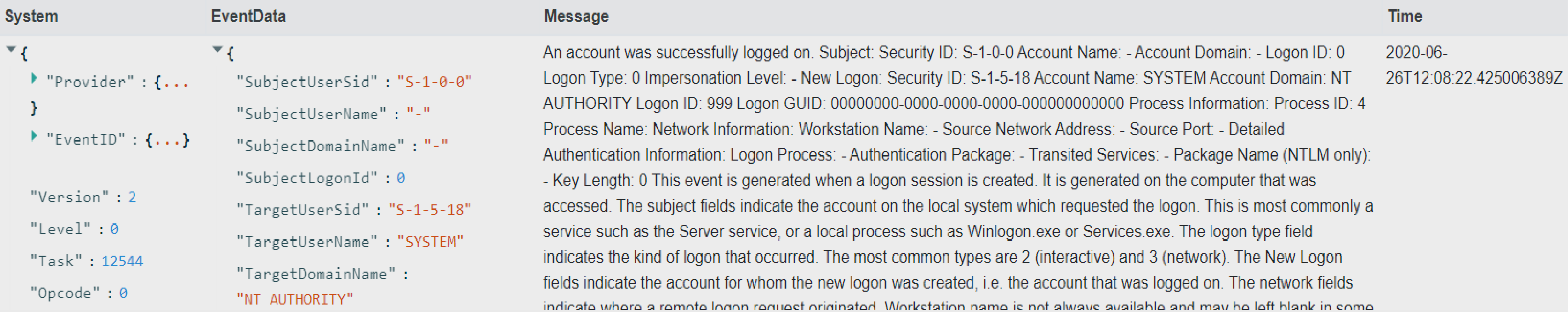
State	FlowId	Artifacts	Created	Last Active	Creator	Mb	Rows
✓	F.C2N32VR200MQC	Windows.Search.Yara	2021-05-26 11:31:11 UTC	2021-05-26 11:32:13 UTC	admin		2

Rule	HitOffset	HitContext	FileName	Size	ModTime	Upload
dtect	0	Secret_Name	C:/Users/Public/Downloads/test.txt	11	2021-05-26T10:14:05.3103753Z	
dtect	0	Secret_Name	C:/Users/monty/test.txt	11	2021-05-26T09:46:08.7705811Z	

Logs

Logs Example

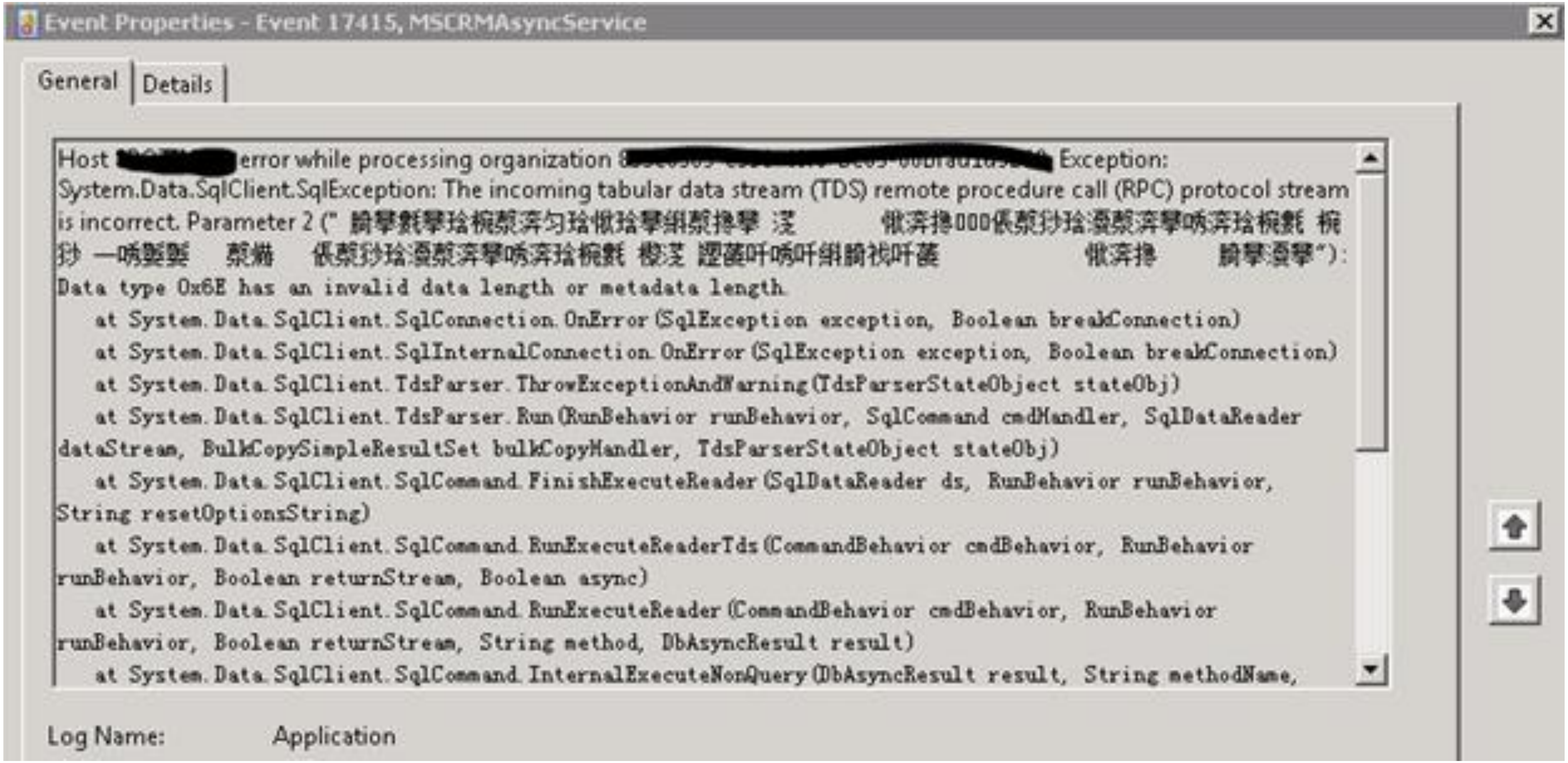
```
LET seclogs <= SELECT FullPath FROM
glob(globs="C:/Windows/System32/winevt/Logs/*Security*.evtx") LIMIT 3

SELECT *, timestamp(epoch=System.TimeCreated.SystemTime) as Time FROM
parse_evtx(filename=seclogs, accessor="ntfs") WHERE System.EventID.Value = 4624
ORDER BY Time
```



System	EventData	Message	Time

{
"Provider" : {...
}
"EventID" : {...}
"Version" : 2
"Level" : 0
"Task" : 12544
"Opcode" : 0
	{
	"SubjectUserSid" : "S-1-0-0"
	"SubjectUserName" : "-"
	"SubjectDomainName" : "-"
	"SubjectLogonId" : 0
	"TargetUserSid" : "S-1-5-18"
	"TargetUserName" : "SYSTEM"
	"TargetDomainName" :
	"NT AUTHORITY"
		An account was successfully logged on. Subject: Security ID: S-1-0-0 Account Name: - Account Domain: - Logon ID: 0 Logon Type: 0 Impersonation Level: - New Logon: Security ID: S-1-5-18 Account Name: SYSTEM Account Domain: NT AUTHORITY Logon ID: 999 Logon GUID: 00000000-0000-0000-0000-000000000000 Process Information: Process ID: 4 Process Name: Network Information: Workstation Name: - Source Network Address: - Source Port: - Detailed Authentication Information: Logon Process: - Authentication Package: - Transited Services: - Package Name (NTLM only): - Key Length: 0 This event is generated when a logon session is created. It is generated on the computer that was accessed. The subject fields indicate the account on the local system which requested the logon. This is most commonly a service such as the Server service, or a local process such as Winlogon.exe or Services.exe. The logon type field indicates the kind of logon that occurred. The most common types are 2 (interactive) and 3 (network). The New Logon fields indicate the account for whom the new logon was created, i.e. the account that was logged on. The network fields indicate where a remote logon request originated. Workstation name is not always available and may be left blank in some	2020-06-26T12:08:22.425006389Z

The Mystery of Chinese Event Log Entries



```
Host [redacted] error while processing organization [redacted] Exception:
System.Data.SqlClient.SqlException: The incoming tabular data stream (TDS) remote procedure call (RPC) protocol stream
is incorrect. Parameter 2 (" 躺攣戤攣琀椀瀀猀攀匀琀椀攣猀猀攀 渀        愀挀琀000倀猀琀愀瀀猀攀挀昀攀啊挀琀愀椀瀀 椀
琀 一啊犀犀    猀攀    倀猀琀愀瀀猀攀挀昀攀啊挀琀愀椀瀀 樀渀 匀攀吀啊吀猀躺琀吀攀            挀琀愀      躺攣瀀攣"):
Data type 0x6E has an invalid data length or metadata length.
   at System.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection)
   at System.Data.SqlClient.SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection)
   at System.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj)
   at System.Data.SqlClient.TdsParser.Run(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader
dataStream, BulkCopySimpleResultSet bulkCopyHandler, TdsParserStateObject stateObj)
   at System.Data.SqlClient.SqlCommand.FinishExecuteReader(SqlDataReader ds, RunBehavior runBehavior,
String resetOptionsString)
   at System.Data.SqlClient.SqlCommand.RunExecuteReaderTds(CommandBehavior cmdBehavior, RunBehavior
runBehavior, Boolean returnStream, Boolean async)
   at System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior
runBehavior, Boolean returnStream, String method, DbAsyncResult result)
   at System.Data.SqlClient.SqlCommand.InternalExecuteNonQuery(DbAsyncResult result, String methodName,
```

Log Name: Application

Conversion Example, Little Endian => Big Endian

```
using System;
using System.Text;

namespace LEBEconversion
{
    class Program
    {
        static void Main(string[] args)
        {
            var source = "䐀攀氀攀琀椀漀渀匀琀愀琀攀";
            var bytes = Encoding.BigEndianUnicode.GetBytes(source);
            var result = Encoding.Unicode.GetString(bytes);
            Console.WriteLine(result);
        }
    }
}
```

Source https://dylanbeattie.net/2016/10/25/the-mystery-of-chinese-junk.html

Dead Disk Forensics

Have a vmdk or dd which you’d like to run a few artefacts on?

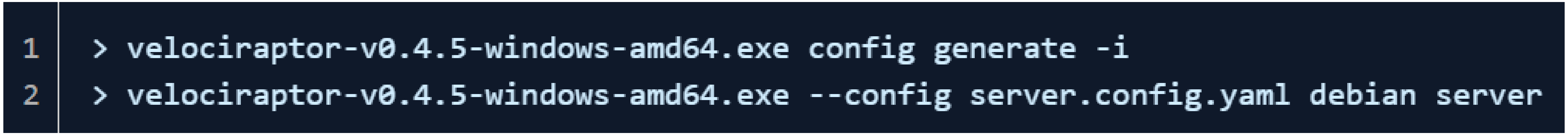
```
// mount vmdk -> creates /mnt/flat
$ vmware-mount -f win10.vmdk /mnt

// auto generate mapping file
// alternatively use disk.dd instead of /mnt/flat
$ velo-linux-amd64 -v deaddisk --add_windows_disk /mnt/flat remapping.yaml

// let’s have a look
$ velo-linux-amd64 --config_override remapping.yaml gui -v
```

Building Velociraptor

Building and Installing Velociraptor

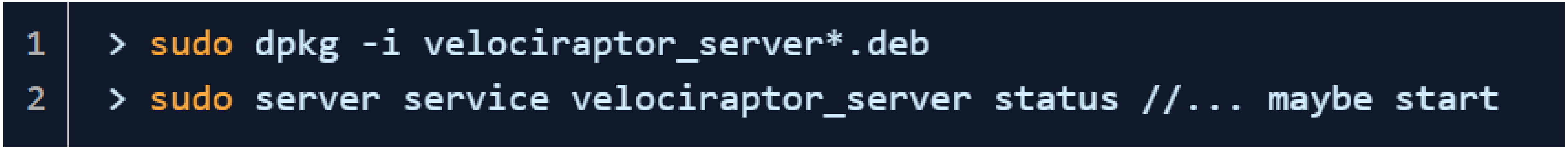
Velociraptor will create configuration files from which you can build clients and servers for various architectures (provided the architecture has a compiler for GO available). Get the latest releases here https://github.com/Velocidex/velociraptor/releases

To build a Linux server Debian installer package on Windows, run



```
> velociraptor-v0.4.5-windows-amd64.exe config generate -i
> velociraptor-v0.4.5-windows-amd64.exe --config server.config.yaml debian server
```

To install the server deb package on Debian, run

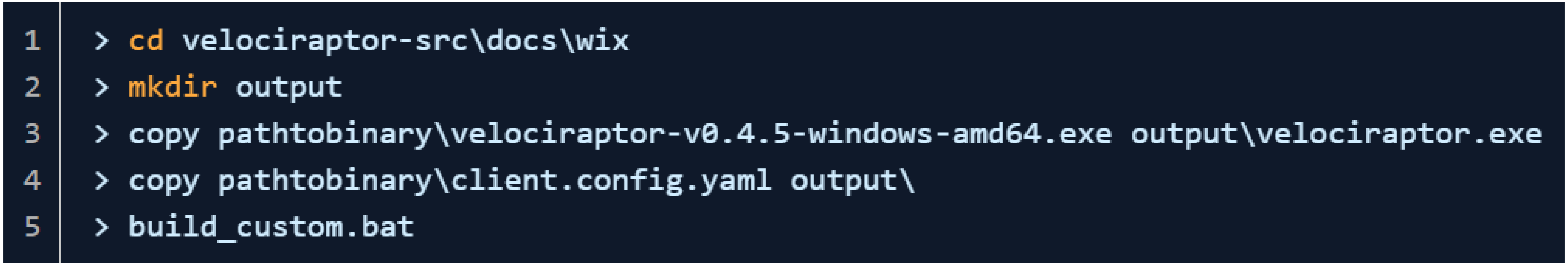


```
> sudo dpkg -i velociraptor_server*.deb
> sudo server service velociraptor_server status //... maybe start
```

Installation of the package might fail due to dependency errors. In that case run **apt-get install -f** and rerun the above.

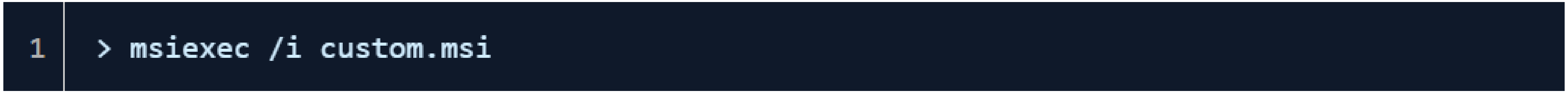
Building a Windows Installer

To create an .msi installer file. Use WIX https://wixtoolset.org/releases/ or unzip the Velociraptor source and change to the WIX folder



```
> cd velociraptor-src\docs\wix
> mkdir output
> copy pathtobinary\velociraptor-v0.4.5-windows-amd64.exe output\velociraptor.exe
> copy pathtobinary\client.config.yaml output\
> build_custom.bat
```

To install the msi Installer file locally, use



```
> msiexec /i custom.msi
```

Windows Deployment

You are strongly advised to sign Windows binaries to avoid hiccups with Defender. Use Microsoft’s signtool.exe or osslsigncode on Linux.

Binaries are usually deployed either by

- GPO scheduled tasks
- GPO assigned software
- Microsoft System Center Configuration Manager (SCCM)
- Some other custom SW deployment mechanism

GPO deployments are fragile. To avoid multiple agents launched you should use the --mutant flag to specify a mutant preventing the agent from starting multiple times.

```
PS C:\> velociraptor.exe --config ... client -v --mutant HoldrioGugu
```

COMPASS
SECURITY