Security Operations With Velociraptor

Eric Capuano – CTO / Co-founder @ Recon InfoSec
@eric_capuano@infosec.exchange

Whitney Champion – Lead Architect / Co-founder @ Recon InfoSec
@shortstack@infosec.exchange

RECON™ INFOSEC

Introduction: Eric Capuano

- CTO / Co-founder @ Recon InfoSec
- SANS DFIR Instructor
- Black Hat Trainer
- Former Cyber Warfare Operator @ Air Nat'l Guard
- Former SOC Manager @ Texas DPS

SANS

OPEN SOC
NETWORK DEFENSE RANGE

black hat

Introduction: Whitney Champion

- Lead Architect / Co-founder @ Recon InfoSec
- Former: Red Hat, Booz Allen Hamilton, SPAWAR
- DEF CON Hacker Tracker
- unicorns.lol

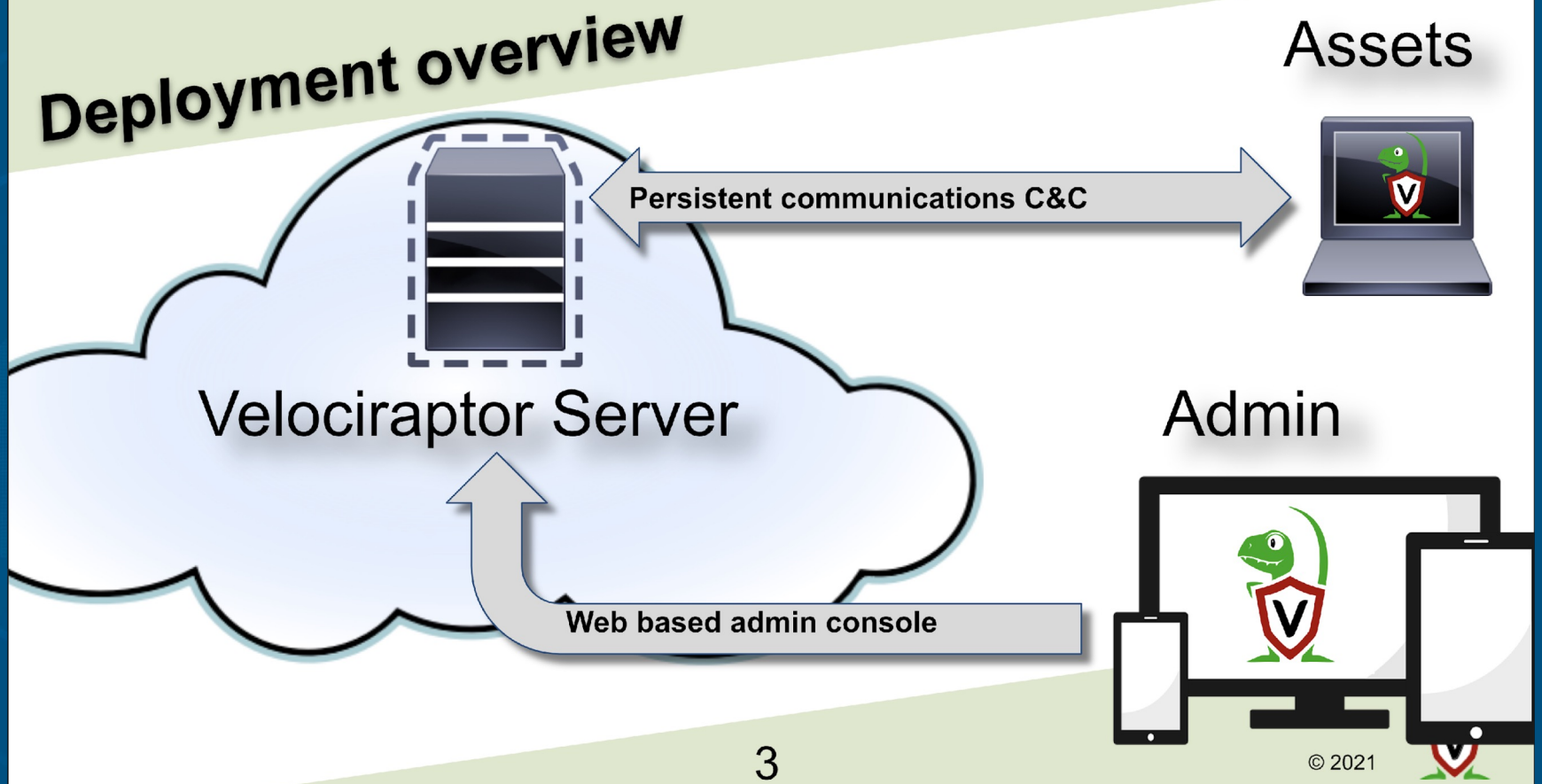
OPEN SOC
NETWORK DEFENSE RANGE

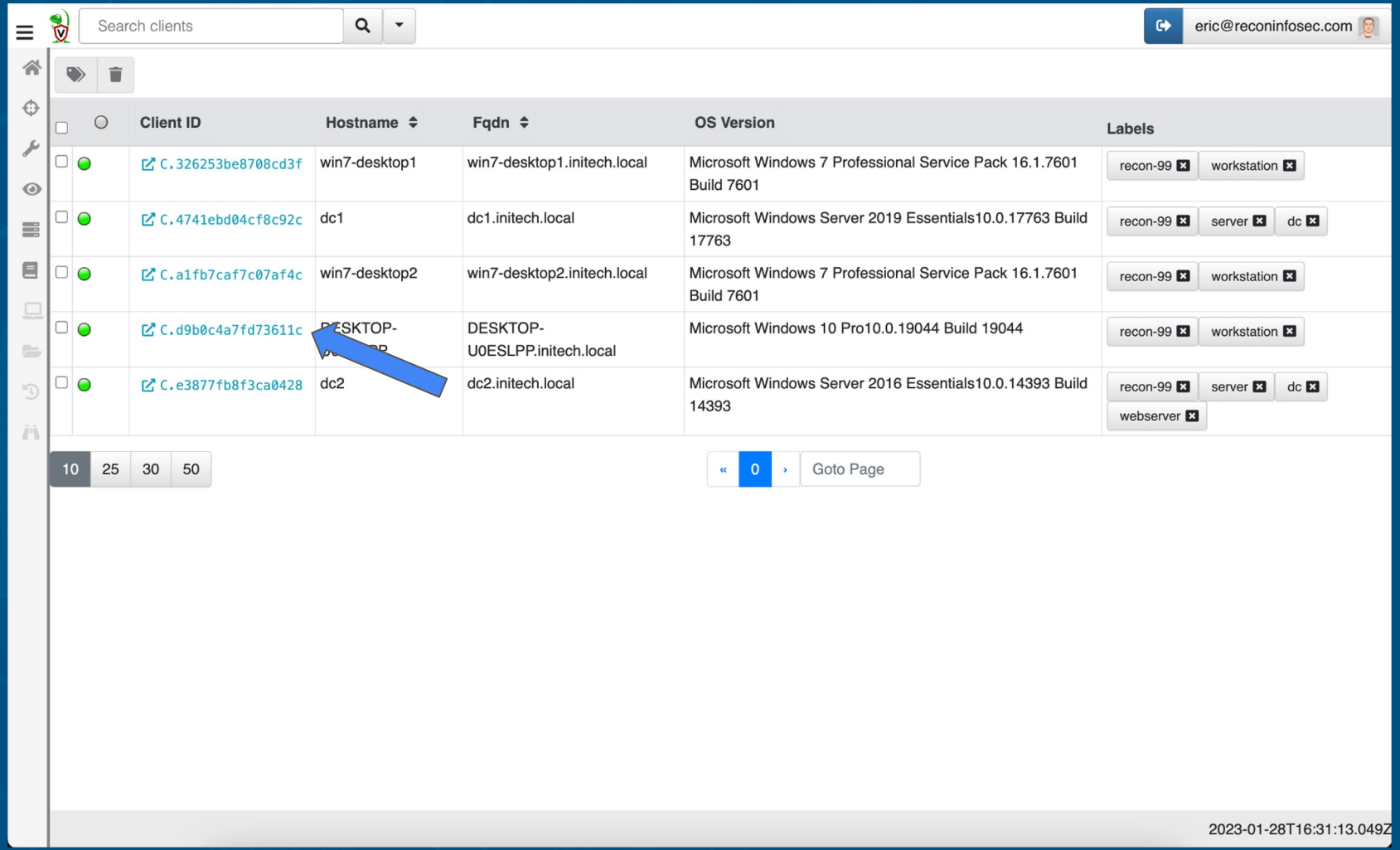
black hat

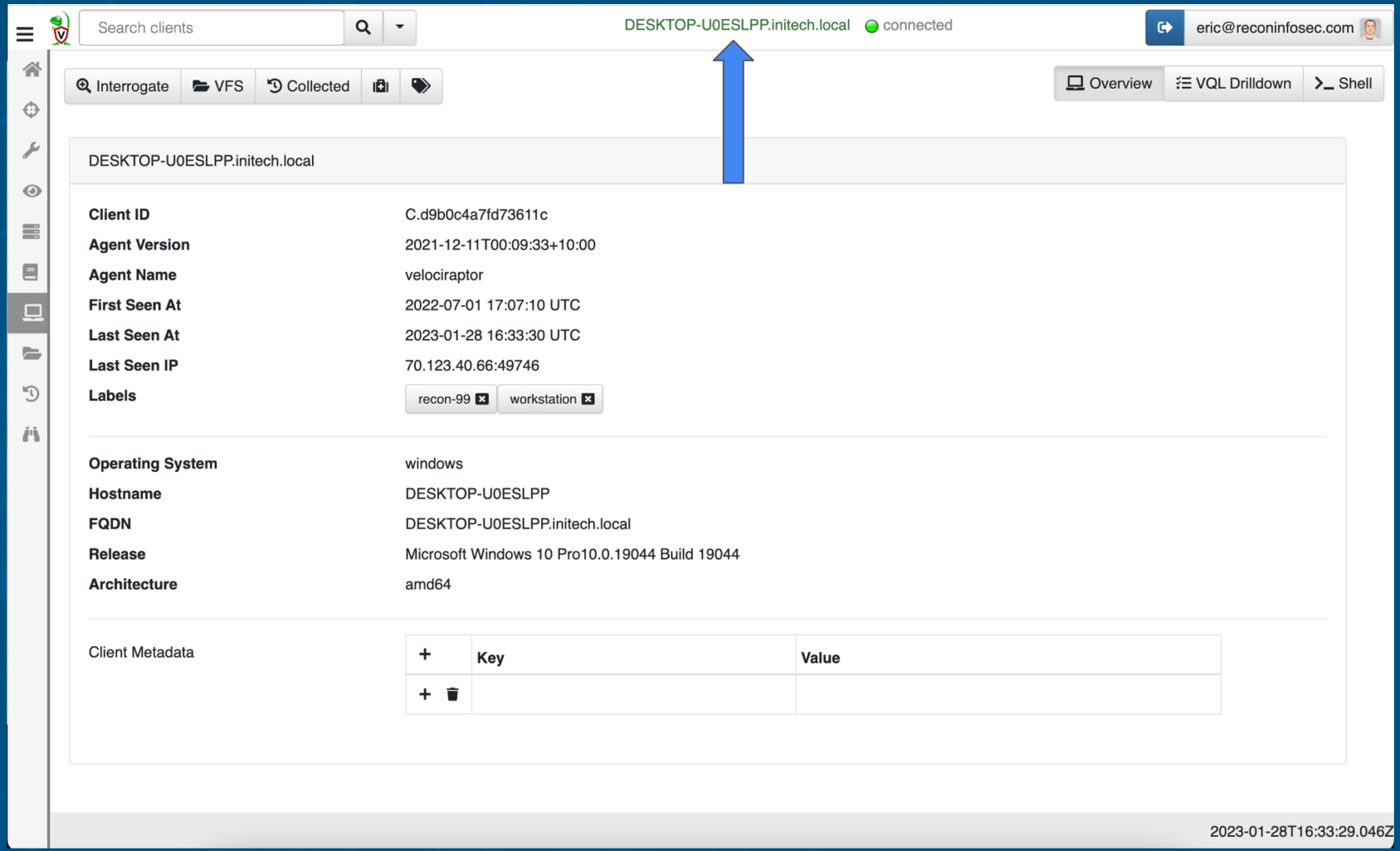
Agenda

- What is Velociraptor? (your new most powerful tool)
- Security Agent Deployment & Monitoring
- Scheduled Hunts & Server Events
- Endpoint Visibility & Interrogation
- Real-Time Eventing & Log Forwarding
- Threat Hunting & Detection
- Incident Response, Containment, & Remediation

Deployment overview

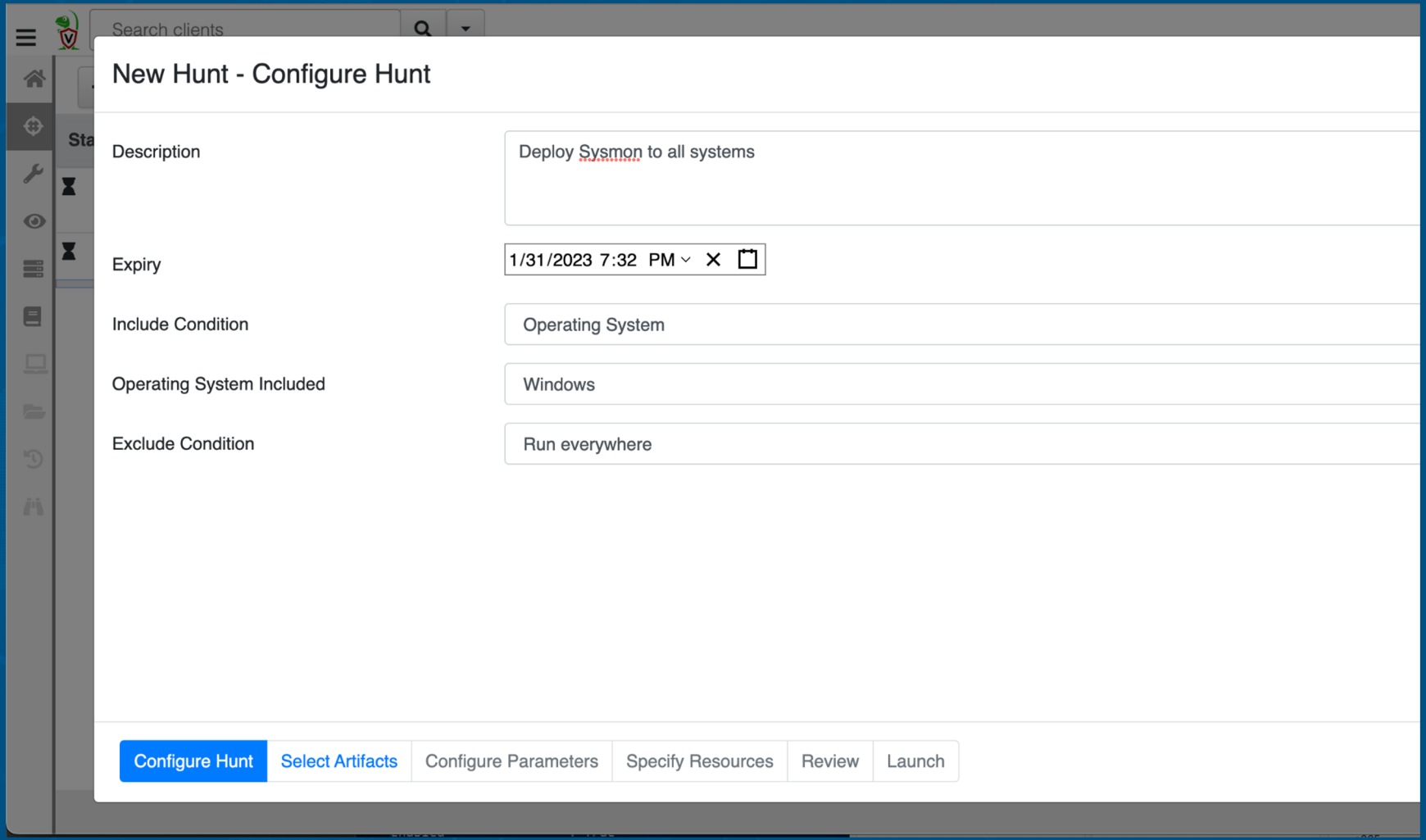


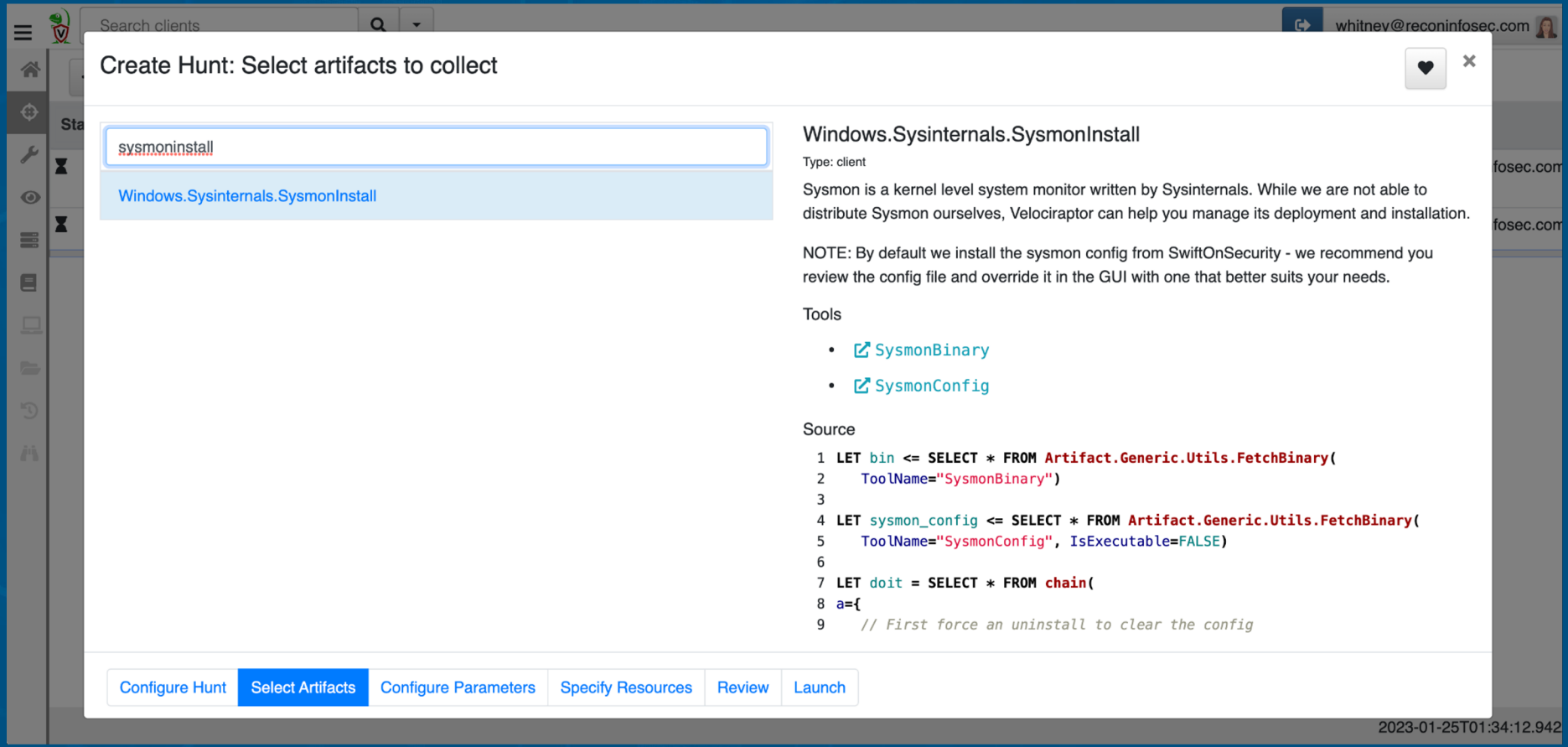

Search clients
eric@reconinfosec.com
Client ID
Hostname
Fqdn
OS Version
Labels
C.326253be8708cd3f
win7-desktop1
win7-desktop1.initech.local
Microsoft Windows 7 Professional Service Pack 16.1.7601 Build 7601
recon-99
workstation
C.4741ebd04cf8c92c
dc1
dc1.initech.local
Microsoft Windows Server 2019 Essentials10.0.17763 Build 17763
recon-99
server
dc
C.a1fb7caf7c07af4c
win7-desktop2
win7-desktop2.initech.local
Microsoft Windows 7 Professional Service Pack 16.1.7601 Build 7601
recon-99
workstation
C.d9b0c4a7fd73611c
DESKTOP-U0ESLPP.initech.local
Microsoft Windows 10 Pro10.0.19044 Build 19044
recon-99
workstation
C.e3877fb8f3ca0428
dc2
dc2.initech.local
Microsoft Windows Server 2016 Essentials10.0.14393 Build 14393
recon-99
server
dc
webserver
10
25
30
50
«
0
›
Goto Page
2023-01-28T16:31:13.049Z


Search clients
DESKTOP-U0ESLPP.initech.local
connected
eric@reconinfosec.com
Interrogate
VFS
Collected
Overview
VQL Drilldown
Shell
DESKTOP-U0ESLPP.initech.local
Client ID
C.d9b0c4a7fd73611c
Agent Version
2021-12-11T00:09:33+10:00
Agent Name
velociraptor
First Seen At
2022-07-01 17:07:10 UTC
Last Seen At
2023-01-28 16:33:30 UTC
Last Seen IP
70.123.40.66:49746
Labels
recon-99
workstation
Operating System
windows
Hostname
DESKTOP-U0ESLPP
FQDN
DESKTOP-U0ESLPP.initech.local
Release
Microsoft Windows 10 Pro10.0.19044 Build 19044
Architecture
amd64
Client Metadata
Key
Value
2023-01-28T16:33:29.046Z

Security Agent Deployment & Monitoring

RECON™ INFOSEC


Search clients
New Hunt - Configure Hunt
Description
Deploy Sysmon to all systems
Expiry
1/31/2023 7:32 PM
Include Condition
Operating System
Operating System Included
Windows
Exclude Condition
Run everywhere
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch


Search clients
whitney@reconinfosec.com
Create Hunt: Select artifacts to collect
sysmoninstall
Windows.Sysinternals.SysmonInstall
Windows.Sysinternals.SysmonInstall
Type: client
Sysmon is a kernel level system monitor written by Sysinternals. While we are not able to distribute Sysmon ourselves, Velociraptor can help you manage its deployment and installation.
NOTE: By default we install the sysmon config from SwiftOnSecurity - we recommend you review the config file and override it in the GUI with one that better suits your needs.
Tools
SysmonBinary
SysmonConfig
Source
1 LET bin <= SELECT * FROM Artifact.Generic.Utils.FetchBinary(
2 ToolName="SysmonBinary")
3
4 LET sysmon_config <= SELECT * FROM Artifact.Generic.Utils.FetchBinary(
5 ToolName="SysmonConfig", IsExecutable=FALSE)
6
7 LET doit = SELECT * FROM chain(
8 a={
9 // First force an uninstall to clear the config
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch
2023-01-25T01:34:12.942

Tool SysmonBinary

Tool Name	SysmonBinary
Upstream URL	https://live.sysinternals.com/tools/sysmon64.exe
Enpoint Filename	sysmon64.exe
Hash	7abcc0edb4c0f9f47a1bac0d06401504e1e91c4b1a4e679f01ad539f364d6881
Serve Locally	
Serve URL	

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 | Click to upload file | Select file

Set Serve URL

Served Locally

Tool will be served from the Velociraptor server to clients if needed. The client will cache the tool on its own disk and compare the hash next time it is needed. Tools will only be downloaded if their hash has changed.

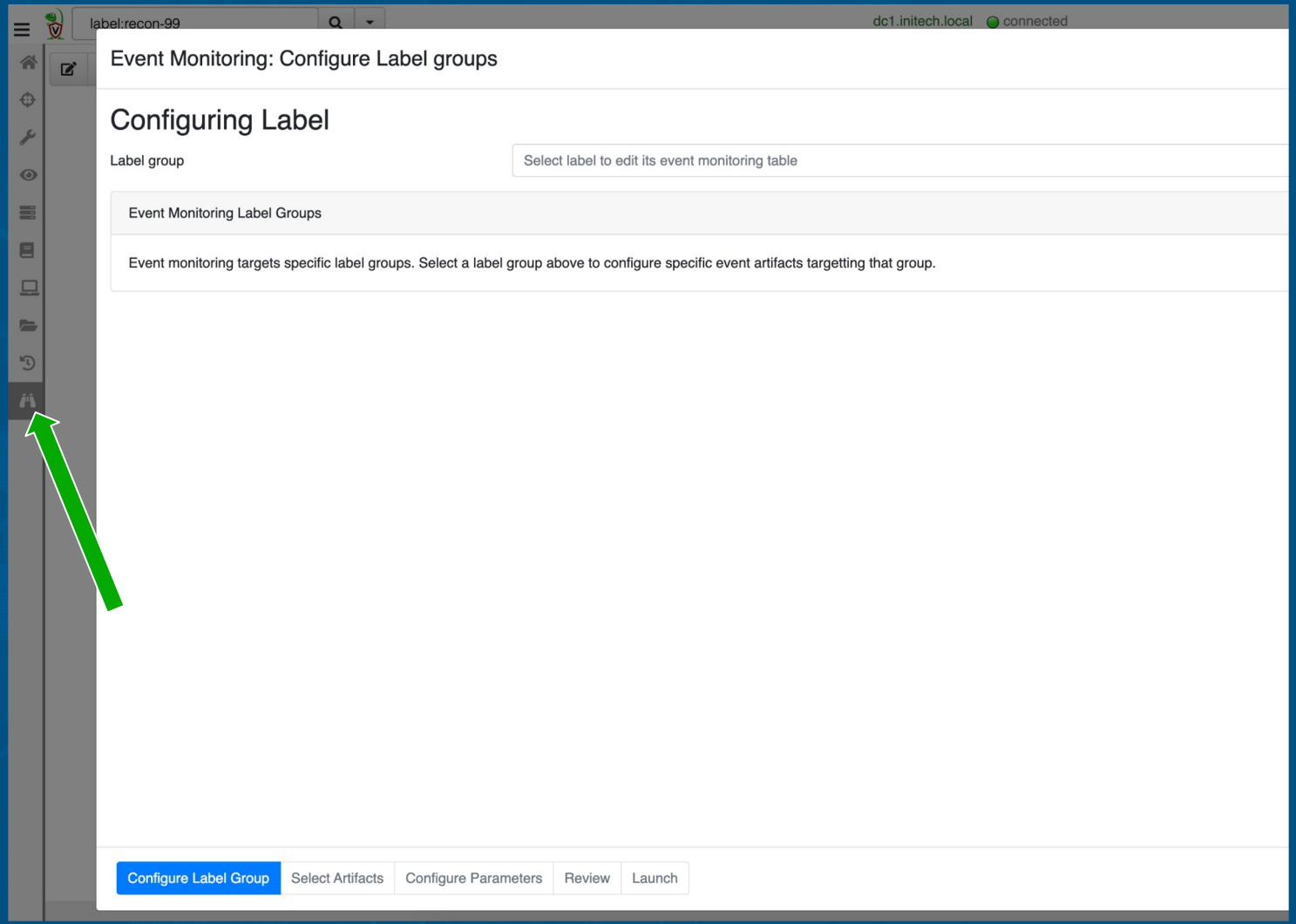
Serve from upstream

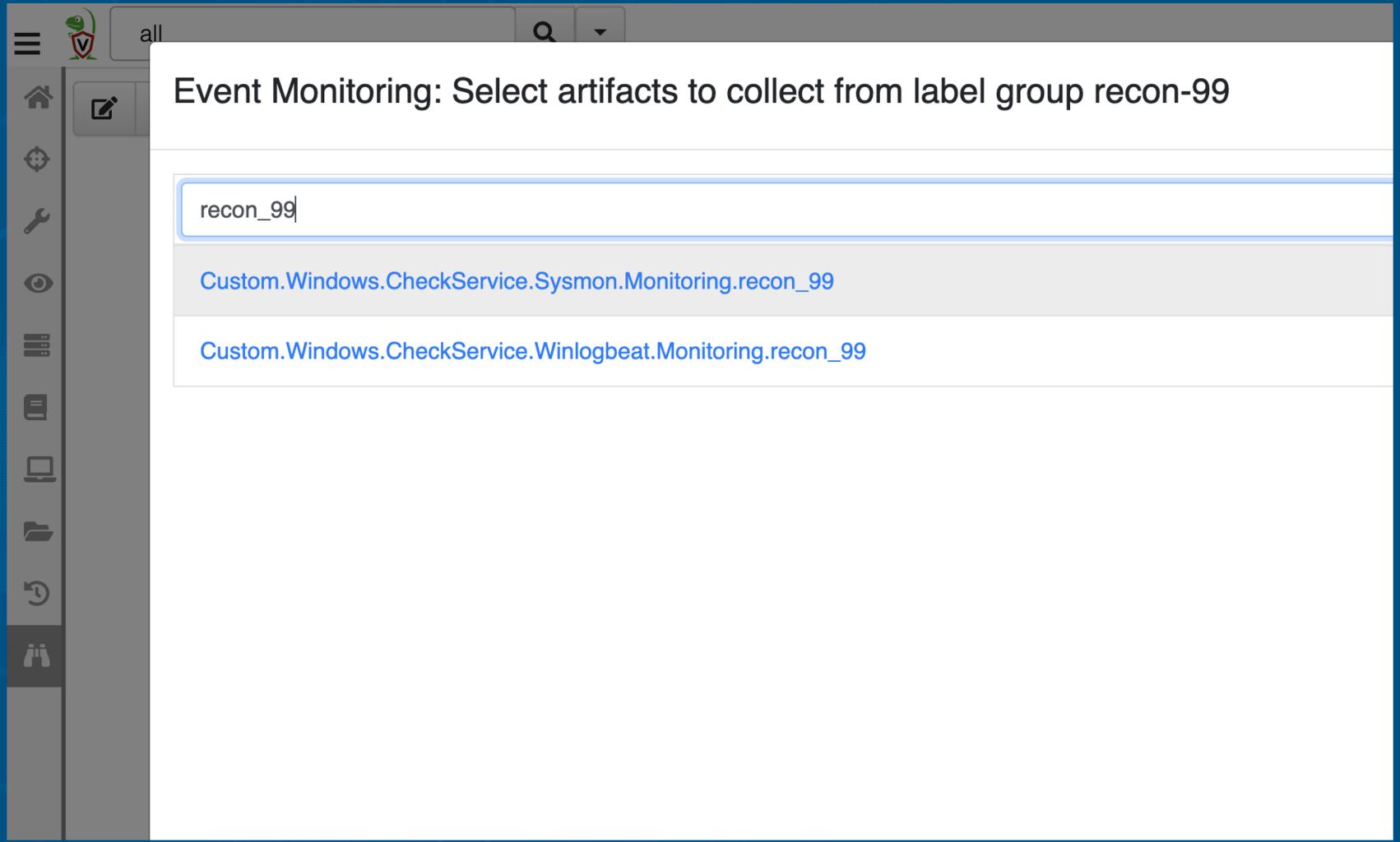
Tool Hash Known

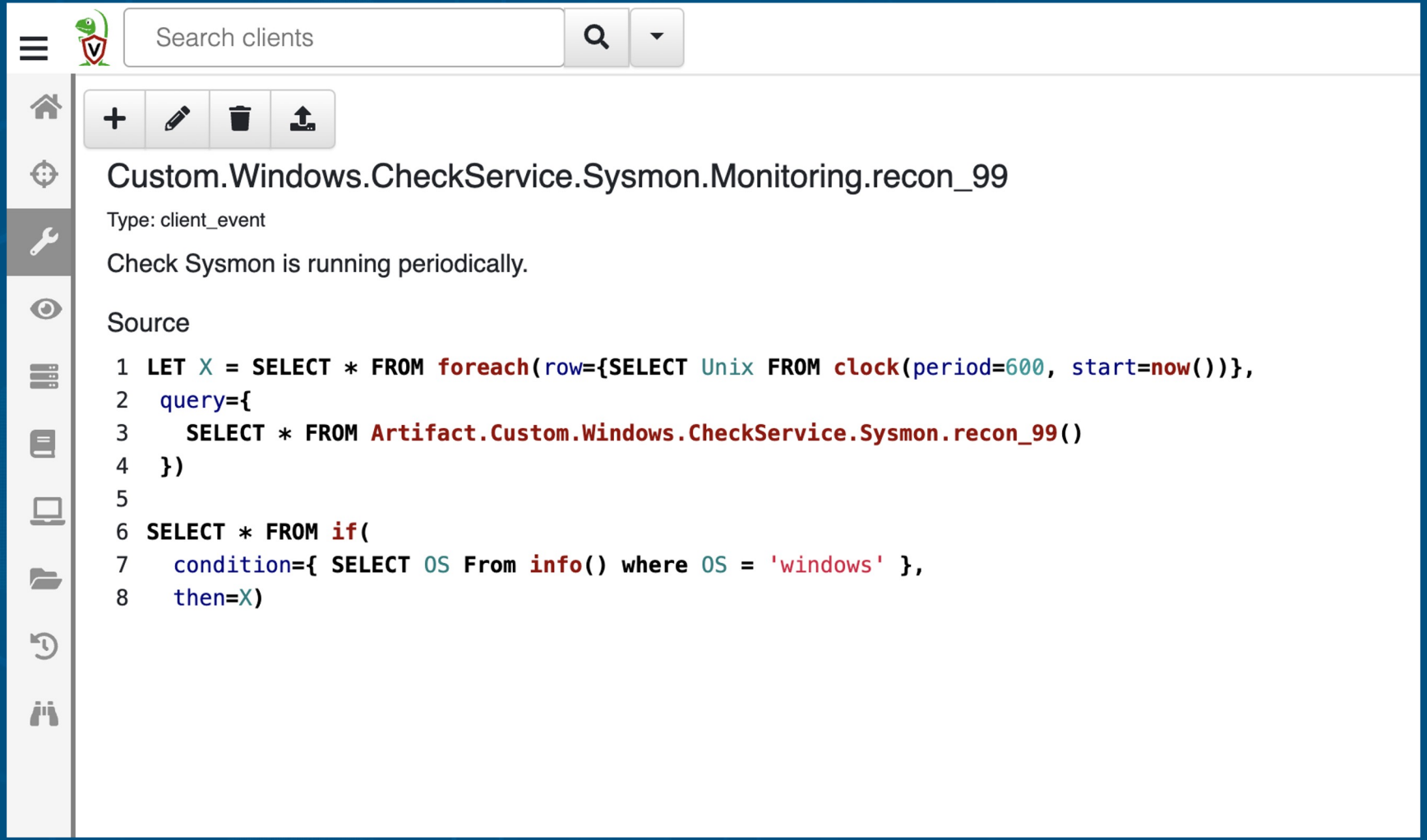
Tool hash has been calculated. When clients need to use this tool they will ensure this hash matches what they download.

Re-Download File

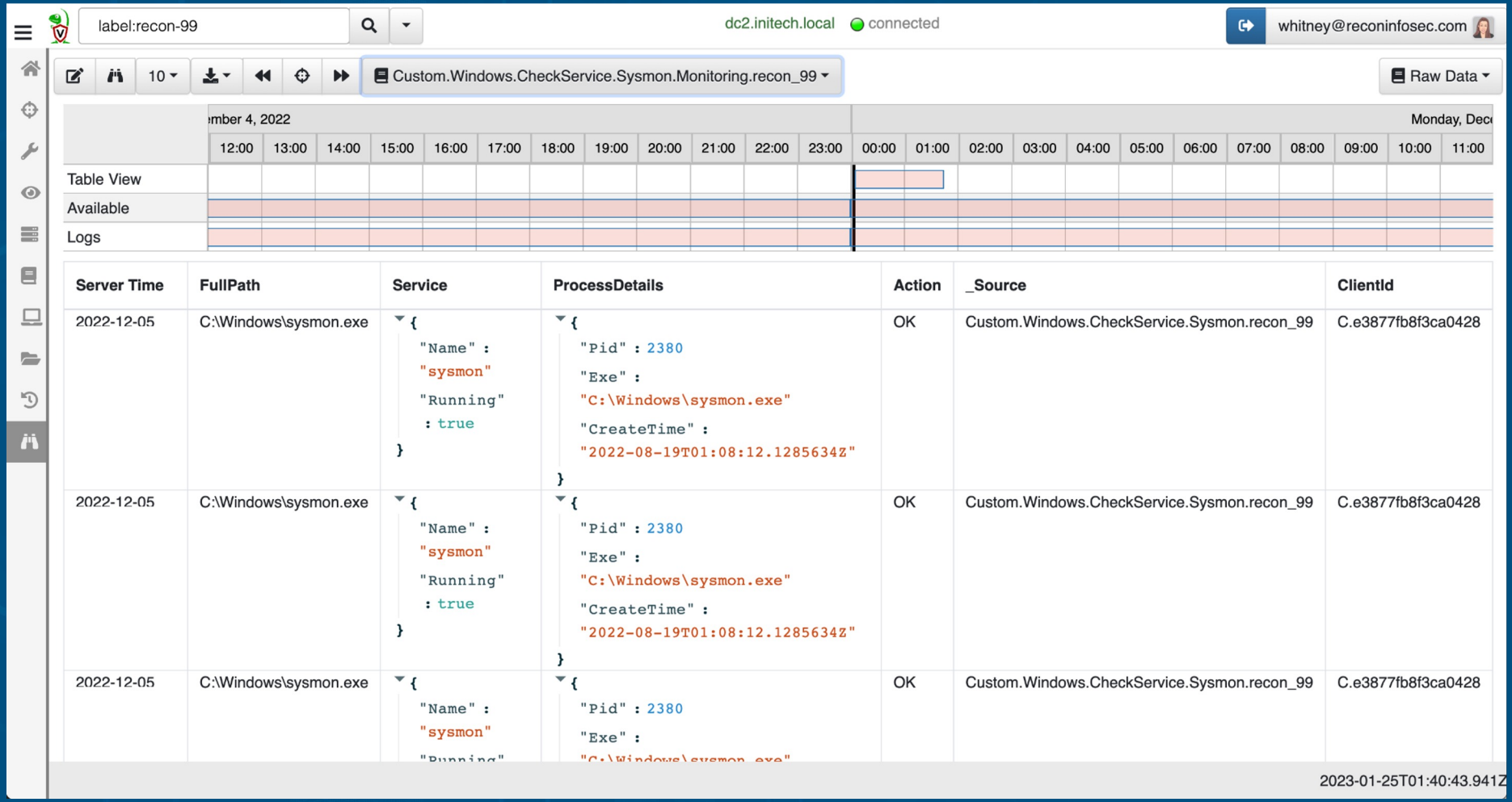
Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch


label:recon-99
dc1.initech.local
connected
Event Monitoring: Configure Label groups
Configuring Label
Label group
Select label to edit its event monitoring table
Event Monitoring Label Groups
Event monitoring targets specific label groups. Select a label group above to configure specific event artifacts targetting that group.
Configure Label Group
Select Artifacts
Configure Parameters
Review
Launch

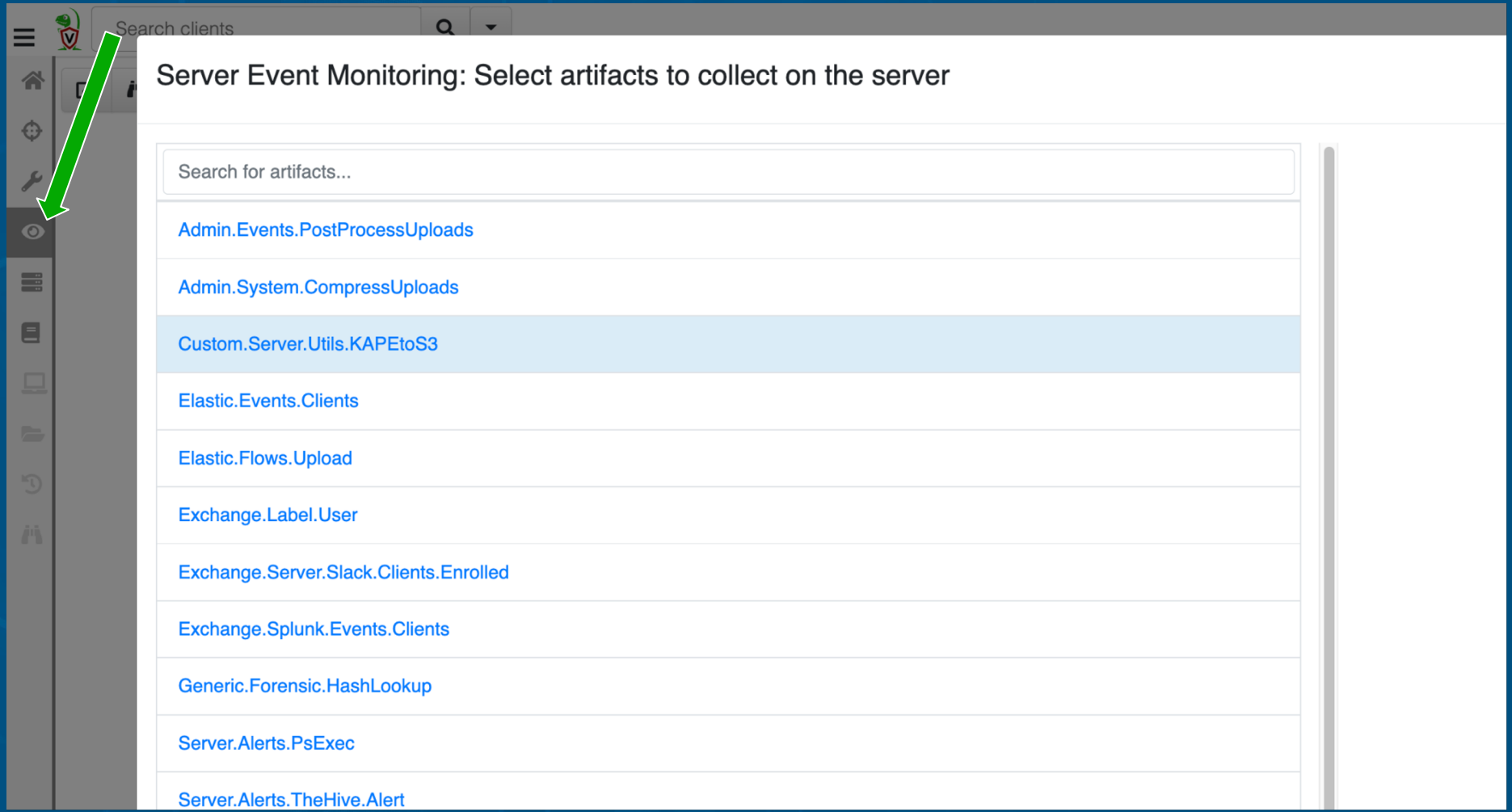

all
Event Monitoring: Select artifacts to collect from label group recon-99
recon_99
Custom.Windows.CheckService.Sysmon.Monitoring.recon_99
Custom.Windows.CheckService.Winlogbeat.Monitoring.recon_99


Search clients
Custom.Windows.CheckService.Sysmon.Monitoring.recon_99
Type: client_event
Check Sysmon is running periodically.
Source
1 LET X = SELECT * FROM foreach(row={SELECT Unix FROM clock(period=600, start=now())},
2 query={
3 SELECT * FROM Artifact.Custom.Windows.CheckService.Sysmon.recon_99()
4 })
5
6 SELECT * FROM if(
7 condition={ SELECT OS From info() where OS = 'windows' },
8 then=X)

```
LET FileExists(pathname) = SELECT FullPath FROM glob(globs=pathname)

LET ServiceDetails(service_name) = SELECT Name, State =~ "Running" AS Running
  FROM wmi(query="SELECT * FROM win32_service WHERE Name like \"%" + service_name + "%\"")
  WHERE Name =~ service_name

LET ProcessDetails(process_name) = SELECT Pid, Exe, CreateTime
  FROM pslist() WHERE  Name =~ process_name

LET Details <= SELECT FileExists(pathname=sysmon_path)[0].FullPath AS FullPath,
       ServiceDetails(service_name=sysmon_service_name)[0] AS Service,
       ProcessDetails(process_name=sysmon_process_name)[0] AS ProcessDetails
FROM scope()

SELECT * FROM foreach(row=Details,
query={
    SELECT * FROM switch(
    // If the service is not installed at all then install it
    a={
        SELECT FullPath, Service, ProcessDetails, "Installing Service" AS Action FROM if(
           condition=NOT FullPath OR NOT Service.Name,
        then={
            // Change this to the name of the installer artifact.
            SELECT * FROM Artifact.Custom.Artifact.Sysmon()
        })
    },
    // Service is installed but the process is not running - start the process
    c={
        SELECT FullPath, Service, ProcessDetails, "Starting Service" AS Action FROM if(
           condition=NOT ProcessDetails.Exe OR NOT Service.Running,
        then={
            SELECT * FROM execve(argv=["sc","start", sysmon_service_name])
        })
    },
    // If we get here - all is well with the world!
    z={
       SELECT FullPath, Service, ProcessDetails, "OK" AS Action
       FROM scope()
    })
})
```


label:recon-99
dc2.initech.local
connected
whitney@reconinfosec.com
10
Custom.Windows.CheckService.Sysmon.Monitoring.recon_99
Raw Data
ember 4, 2022
Monday, Dec
12:00 13:00 14:00 15:00 16:00 17:00 18:00 19:00 20:00 21:00 22:00 23:00 00:00 01:00 02:00 03:00 04:00 05:00 06:00 07:00 08:00 09:00 10:00 11:00
Table View
Available
Logs
Server Time
FullPath
Service
ProcessDetails
Action
_Source
ClientId
2022-12-05
C:\Windows\sysmon.exe
{ "Name" : "sysmon" "Running" : true }
{ "Pid" : 2380 "Exe" : "C:\Windows\sysmon.exe" "CreateTime" : "2022-08-19T01:08:12.1285634Z" }
OK
Custom.Windows.CheckService.Sysmon.recon_99
C.e3877fb8f3ca0428
2022-12-05
C:\Windows\sysmon.exe
{ "Name" : "sysmon" "Running" : true }
{ "Pid" : 2380 "Exe" : "C:\Windows\sysmon.exe" "CreateTime" : "2022-08-19T01:08:12.1285634Z" }
OK
Custom.Windows.CheckService.Sysmon.recon_99
C.e3877fb8f3ca0428
2022-12-05
C:\Windows\sysmon.exe
{ "Name" : "sysmon"
{ "Pid" : 2380 "Exe" :
OK
Custom.Windows.CheckService.Sysmon.recon_99
C.e3877fb8f3ca0428
2023-01-25T01:40:43.941Z

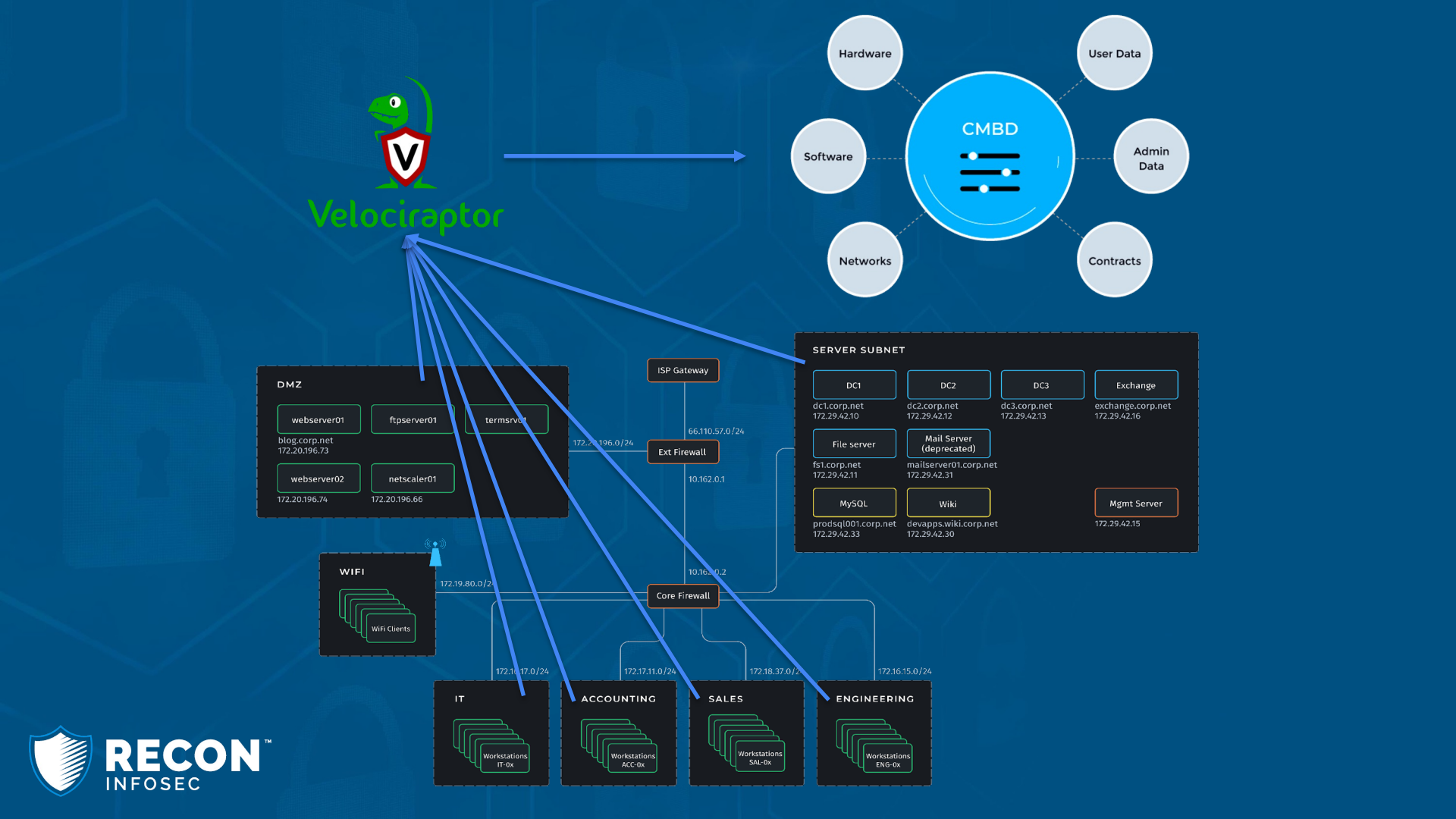
Scheduled Hunts & Server Events


Search clients
Server Event Monitoring: Select artifacts to collect on the server
Search for artifacts...
Admin.Events.PostProcessUploads
Admin.System.CompressUploads
Custom.Server.Utils.KAPEtoS3
Elastic.Events.Clients
Elastic.Flows.Upload
Exchange.Label.User
Exchange.Server.Slack.Clients.Enrolled
Exchange.Splunk.Events.Clients
Generic.Forensic.HashLookup
Server.Alerts.PsExec
Server.Alerts.TheHive.Alert

```
LET upload_to_s3(ClientId, FlowId, Fqdn) = SELECT ClientId,
       upload_s3(bucket=bucket,
                 credentialskey=credentialskey,
                 credentialssecret=credentialssecret,
                 region=region,
                 file=output_file,
                 name=format(format="Host %v %v %v.zip",
                    args=[Fqdn, FlowId, timestamp(epoch=now())])) AS S3
FROM collect(artifacts="UploadFlow", artifact_definitions=UploadFlowDefinition,
             args=dict(`UploadFlow`=dict(
                    ClientId=ClientId, FlowId=FlowId)),
             output=output_file)

LET completions = SELECT *, client_info(client_id=ClientId).os_info.fqdn AS Fqdn
    FROM watch_monitoring(artifact="System.Flow.Completion")
    WHERE Flow.artifacts_with_results =~ ArtifactNameRegex

SELECT * FROM foreach(row=completions, query={
  SELECT * FROM upload_to_s3(ClientId=ClientId, FlowId=FlowId, Fqdn=Fqdn)
})
```

Full artifact available on GitHub:

https://github.com/ReconInfoSec/velociraptor-to-timesketch



RECON INFOSEC

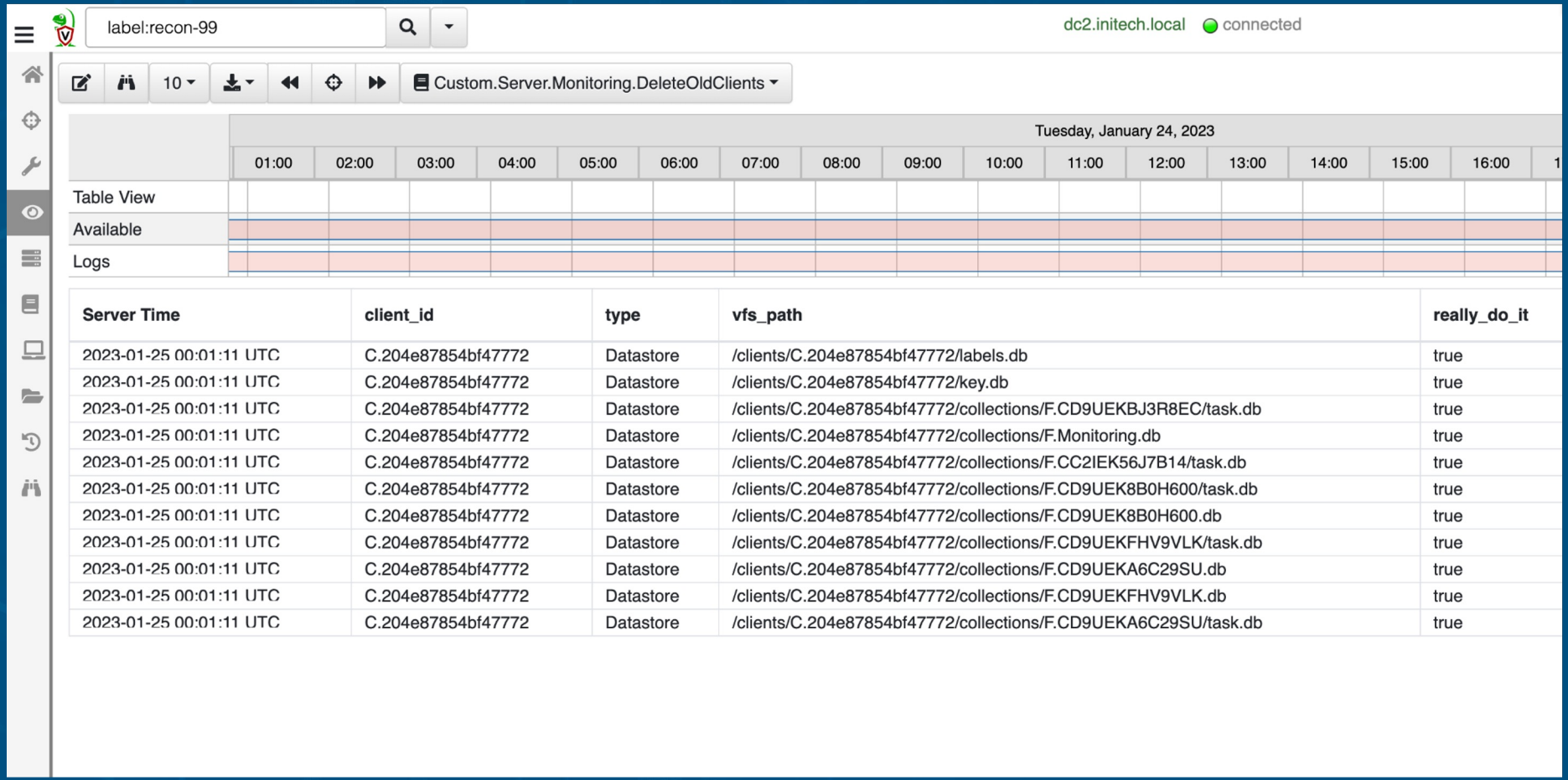
```
name: Custom.Server.Monitoring.Collect.Windows.Services
description: |
     Run artifact weekly on schedule

type: SERVER_EVENT

parameters:
  - name: ScheduleDayRegex
    default: Sunday
  - name: ScheduleTimeRegex
    default: "03:00:"
  - name: HuntDescription
    default:

sources:
  - query: |
      LET schedule = SELECT UTC.String AS Now,
      Weekday.String AS Today
      FROM clock(period=60)
      WHERE Now =~ ScheduleTimeRegex AND Today =~ ScheduleDayRegex AND
          log(message="Launching at time " + Now)

      SELECT hunt(artifacts=["Custom.Windows.Services"],
                  spec=dict(`Custom.Windows.Services`=dict()),
                  description="Weekly hunt - Custom.Windows.Services")
      FROM schedule
```

```
name: Custom.Server.Utils.MySQL
description: |
   This server monitoring artifact will automatically import any collected
   artifacts that match the regex into a SQL database.

type: SERVER_EVENT

parameters:
   - name: ArtifactNameRegex
     default: ""
     description: A regular expression to select which artifacts to upload
   - name: sql_host
     default: ""
   - name: sql_database
     default: ""
   - name: sql_username
     default: ""
   - name: sql_password
     default: ""

sources:
  - query: |
       LET sql_host <= if(condition=sql_host, then=sql_host,
          else=server_metadata().MySQLServer)
       LET sql_password <= if(condition=sql_password, then=sql_password,
          else=server_metadata().MySQLPassword)

       LET completions = SELECT *
           FROM watch_monitoring(artifact="System.Flow.Completion")
           WHERE Flow.artifacts_with_results =~ ArtifactNameRegex

       LET all_results = SELECT * FROM foreach(row=completions,
           query={
             SELECT *, ClientId FROM source(client_id=ClientId, flow_id=FlowId, artifact=Flow.artifacts_with_results[0])
           })

       SELECT column1_value, column2_value, column3_value, column4_value, column5_value,
           {
             SELECT * FROM sql(driver="mysql",
             connstring=sql_username+":"+sql_password+"@tcp("+sql_host+":3306)/"+sql_database,
             query='INSERT IGNORE INTO table_name (column1, column2, column3, column4, column5) VALUES (?,?,?,?,?)',
             args=[column1_value, column2_value, column3_value, column4_value, column5_value])

           } AS Insert FROM all_results
```

Easily collect endpoint data from artifact collections and automagically send it elsewhere for further analysis.

- Elasticsearch
- Splunk
- MySQL
- PostgreSQL
- SQLite
- S3

```
name: Custom.Server.Monitoring.DeleteOldClients
description: |
     Run client delete periodically
type: SERVER_EVENT

parameters:
  - name: ScheduleDayRegex
    default: day
    type: regex
  - name: ScheduleTimeRegex
    default: "00:00:"
    type: regex
  - name: Days
    default: 60
  - name: ReallyDoIt
    type: bool

sources:
  - query: |

        LET old_clients = SELECT os_info.fqdn AS Fqdn, client_id,
                timestamp(epoch=last_seen_at/1000000) AS LastSeen FROM clients()
        WHERE LastSeen < now() - ( atoi(string=Days) * 3600 * 24 ) AND log(message=now())

        LET schedule = SELECT UTC.String AS Now, Weekday.String AS Today
        FROM clock(period=60)
        WHERE Now =~ ScheduleTimeRegex
            AND Today =~ ScheduleDayRegex
            AND log(message="Launching at time " + Now)

        SELECT * FROM foreach(row=schedule,
            query={
                SELECT * FROM foreach(row=old_clients,
                    query={
                        SELECT *, Fqdn, LastSeen FROM client_delete(client_id=client_id, really_do_it=ReallyDoIt)
                    }
                )
        })
```



Server Time	client_id	type	vfs_path	really_do_it
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/labels.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/key.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEKBJ3R8EC/task.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.Monitoring.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CC2IEK56J7B14/task.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEK8B0H600/task.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEK8B0H600.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEKFHV9VLK/task.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEKA6C29SU.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEKFHV9VLK.db	true
2023-01-25 00:01:11 UTC	C.204e87854bf47772	Datastore	/clients/C.204e87854bf47772/collections/F.CD9UEKA6C29SU/task.db	true

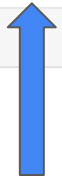
Endpoint Visibility & Interrogation

Search clients

DESKTOP-U0ESLPP.initech.local connected

eric@reconinfosec.com

Interrogate | VFS | Collected | Overview | VQL Drilldown | Shell



DESKTOP-U0ESLPP.initech.local

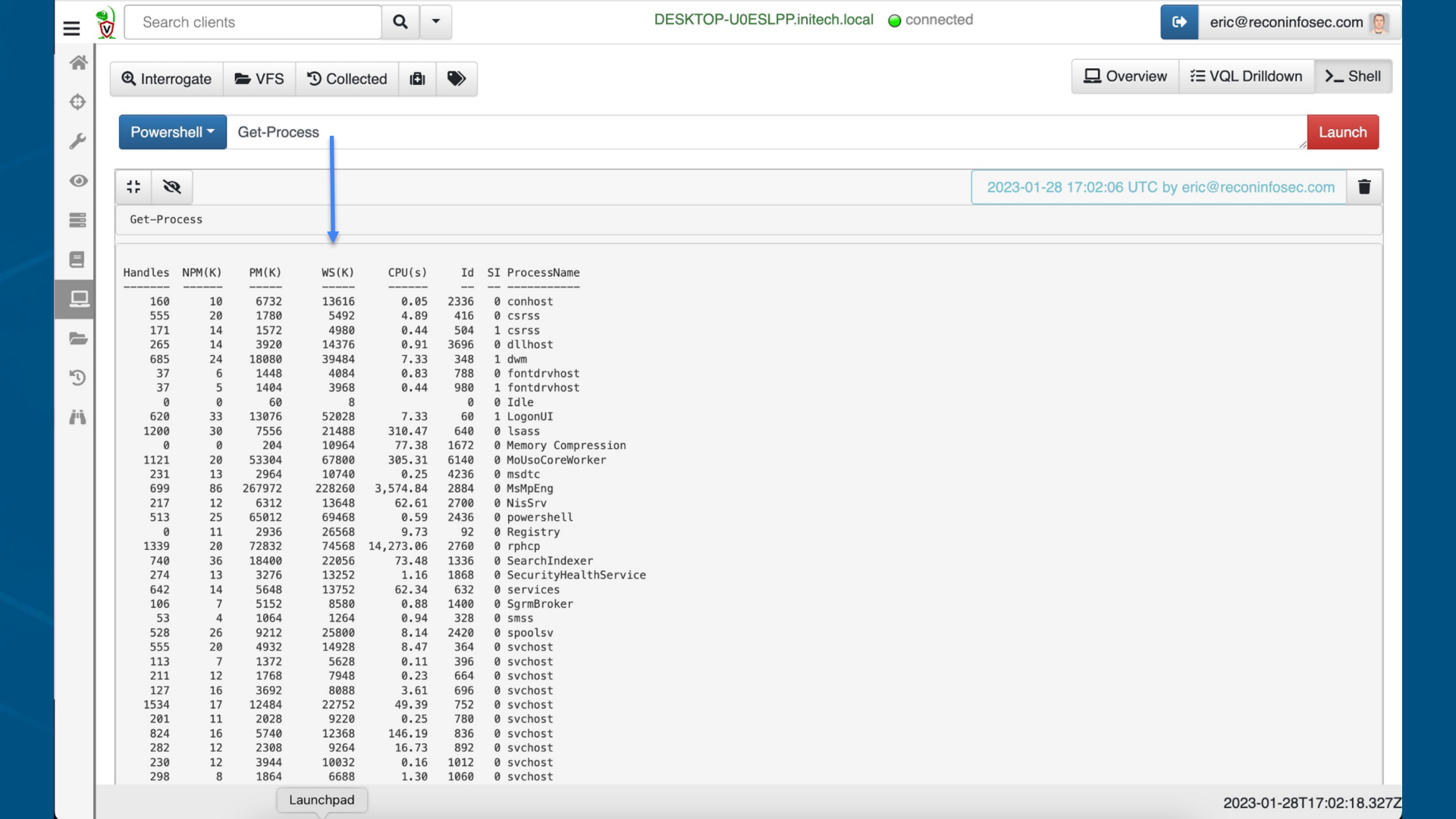
Client ID	C.d9b0c4a7fd73611c
Agent Version	2021-12-11T00:09:33+10:00
Agent Name	velociraptor
First Seen At	2022-07-01 17:07:10 UTC
Last Seen At	2023-01-28 16:33:30 UTC
Last Seen IP	70.123.40.66:49746
Labels	recon-99 workstation

Operating System	windows
Hostname	DESKTOP-U0ESLPP
FQDN	DESKTOP-U0ESLPP.initech.local
Release	Microsoft Windows 10 Pro10.0.19044 Build 19044
Architecture	amd64

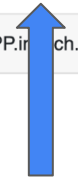
Client Metadata

+	Key	Value
+		

2023-01-28T16:33:29.046Z



Handles	NPM(K)	PM(K)	WS(K)	CPU(s)	Id	SI	ProcessName
160	10	6732	13616	0.05	2336	0	conhost
555	20	1780	5492	4.89	416	0	csrss
171	14	1572	4980	0.44	504	1	csrss
265	14	3920	14376	0.91	3696	0	dllhost
685	24	18080	39484	7.33	348	1	dwm
37	6	1448	4084	0.83	788	0	fontdrvhost
37	5	1404	3968	0.44	980	1	fontdrvhost
0	0	60	8		0	0	Idle
620	33	13076	52028	7.33	60	1	LogonUI
1200	30	7556	21488	310.47	640	0	lsass
0	0	204	10964	77.38	1672	0	Memory Compression
1121	20	53304	67800	305.31	6140	0	MoUsoCoreWorker
231	13	2964	10740	0.25	4236	0	msdtc
699	86	267972	228260	3,574.84	2884	0	MsMpEng
217	12	6312	13648	62.61	2700	0	NisSrv
513	25	65012	69468	0.59	2436	0	powershell
0	11	2936	26568	9.73	92	0	Registry
1339	20	72832	74568	14,273.06	2760	0	rphcp
740	36	18400	22056	73.48	1336	0	SearchIndexer
274	13	3276	13252	1.16	1868	0	SecurityHealthService
642	14	5648	13752	62.34	632	0	services
106	7	5152	8580	0.88	1400	0	SgrmBroker
53	4	1064	1264	0.94	328	0	smss
528	26	9212	25800	8.14	2420	0	spoolsv
555	20	4932	14928	8.47	364	0	svchost
113	7	1372	5628	0.11	396	0	svchost
211	12	1768	7948	0.23	664	0	svchost
127	16	3692	8088	3.61	696	0	svchost
1534	17	12484	22752	49.39	752	0	svchost
201	11	2028	9220	0.25	780	0	svchost
824	16	5740	12368	146.19	836	0	svchost
282	12	2308	9264	16.73	892	0	svchost
230	12	3944	10032	0.16	1012	0	svchost
298	8	1864	6688	1.30	1060	0	svchost



Search clients

DESKTOP-U0ESLPP.initech.local connected

eric@reconinfosec.com

Interrogate | VFS | Collected

Overview | VQL Drilldown | Shell

DESKTOP-U0ESLPP.initech.local

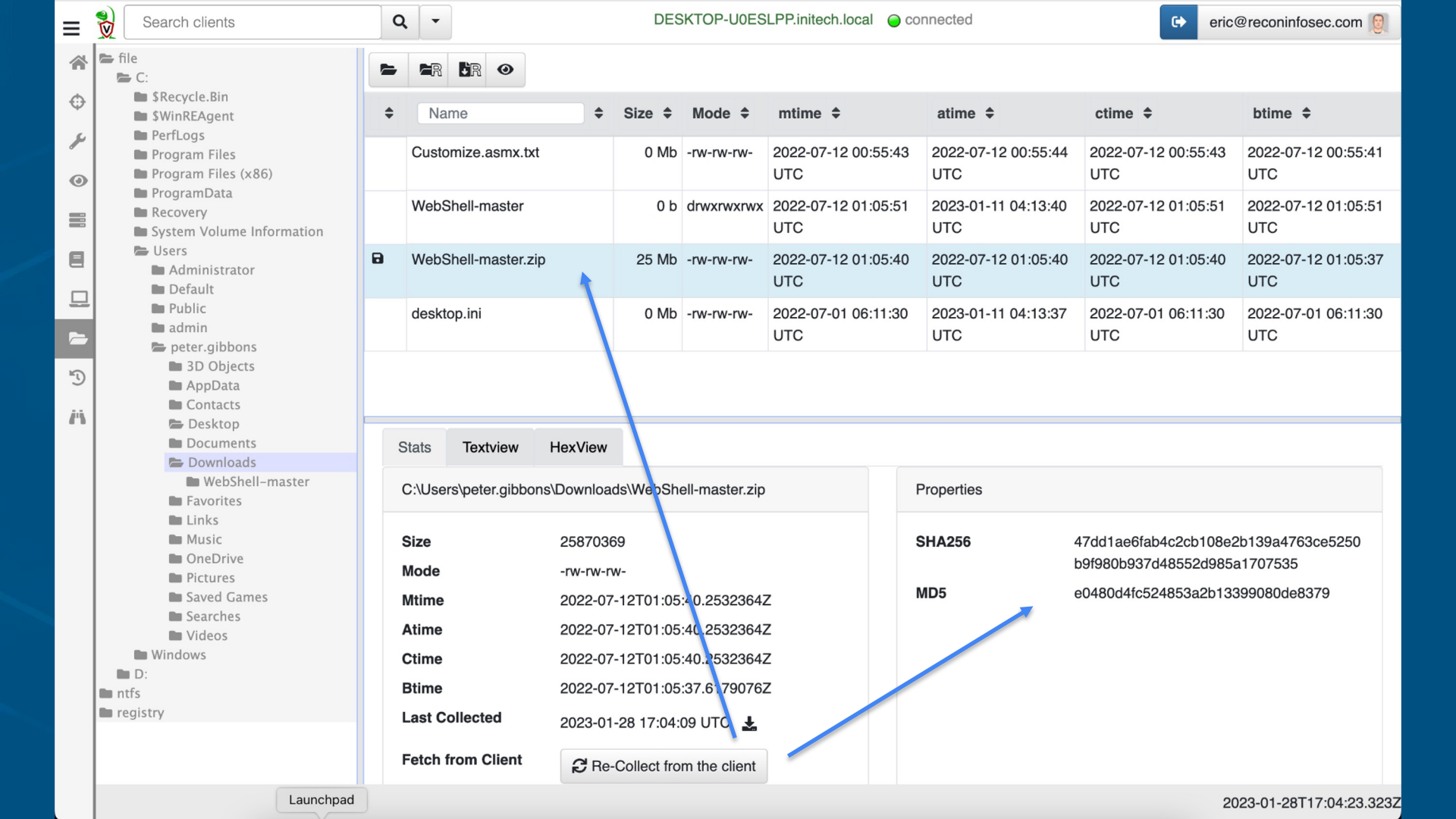
Client ID	C.d9b0c4a7fd73611c
Agent Version	2021-12-11T00:09:33+10:00
Agent Name	velociraptor
First Seen At	2022-07-01 17:07:10 UTC
Last Seen At	2023-01-28 16:33:30 UTC
Last Seen IP	70.123.40.66:49746
Labels	recon-99 workstation

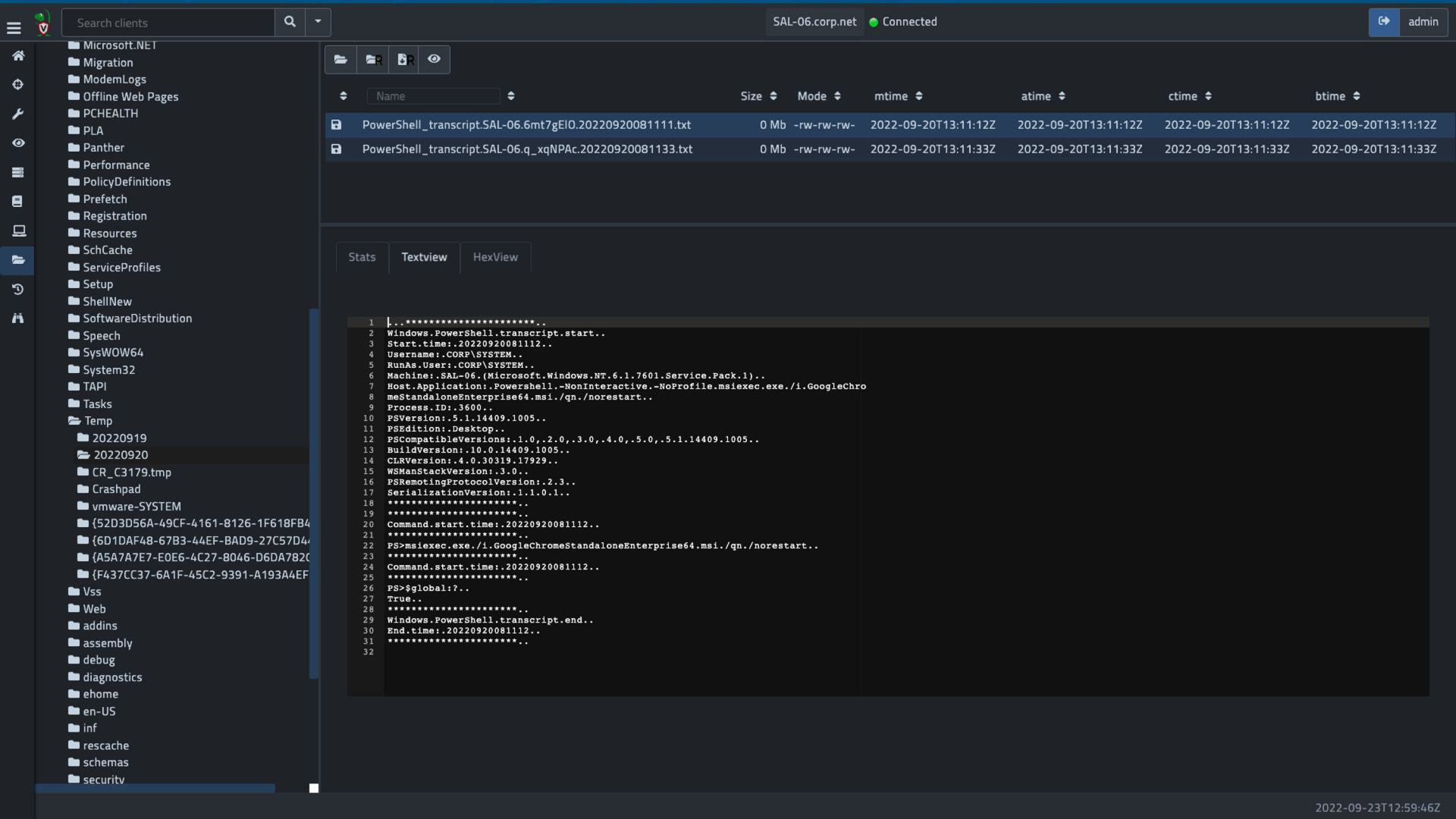
Operating System	windows
Hostname	DESKTOP-U0ESLPP
FQDN	DESKTOP-U0ESLPP.initech.local
Release	Microsoft Windows 10 Pro10.0.19044 Build 19044
Architecture	amd64

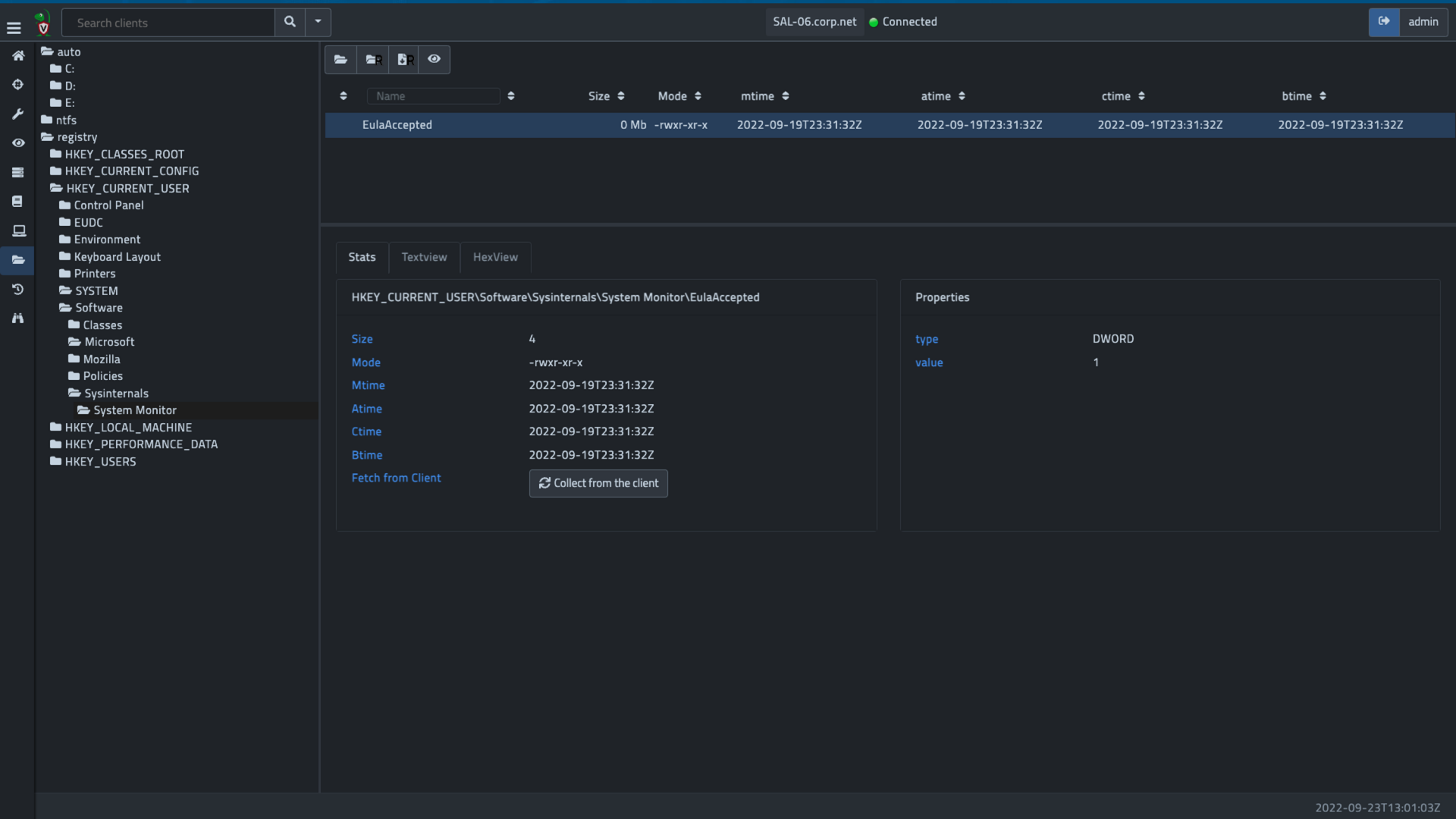
Client Metadata

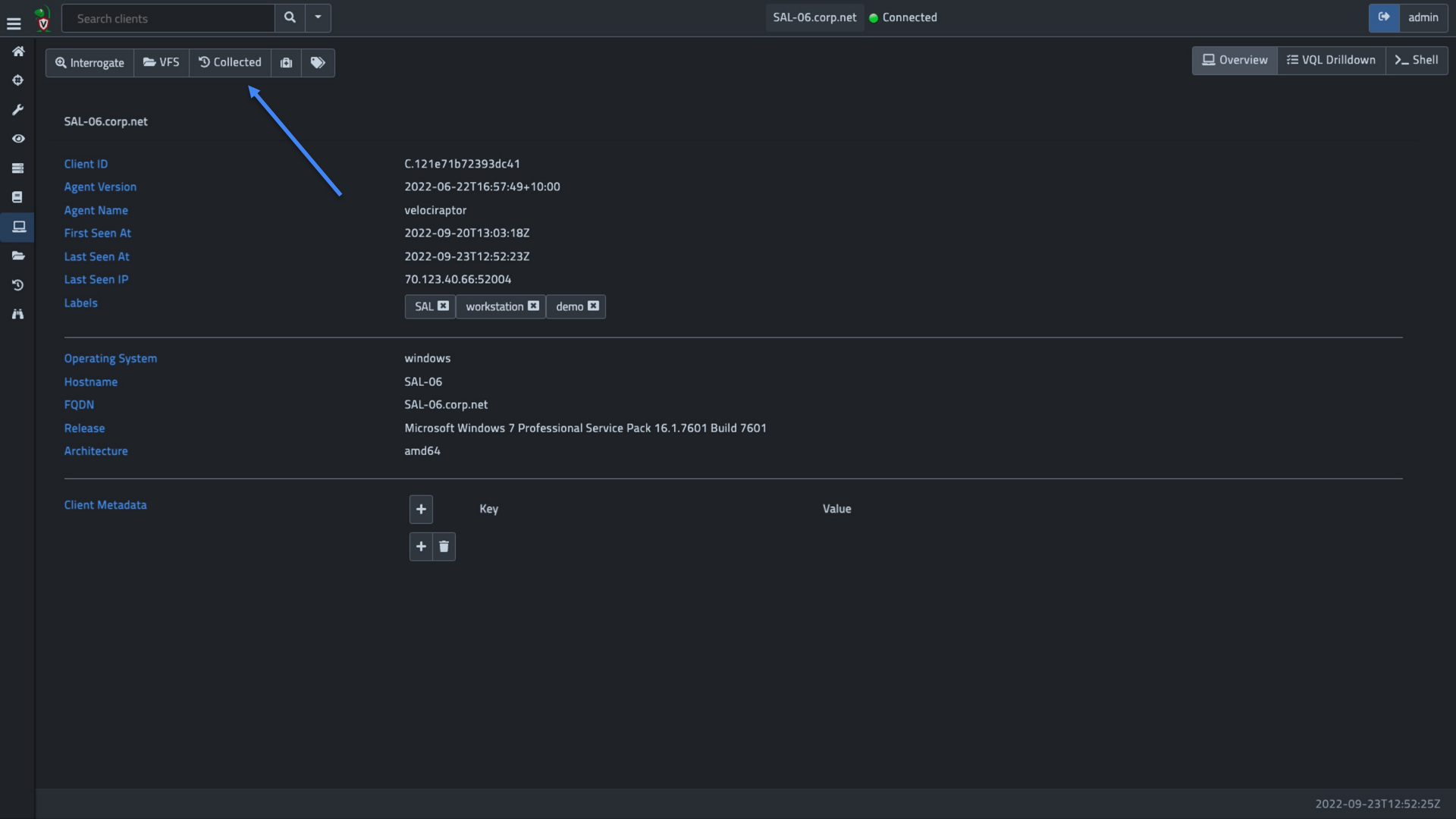
+	Key	Value
+		

2023-01-28T16:33:29.046Z


Search clients
DESKTOP-U0ESLPP.initech.local
connected
eric@reconinfosec.com
file
C:
$Recycle.Bin
$WinREAgent
PerfLogs
Program Files
Program Files (x86)
ProgramData
Recovery
System Volume Information
Users
Administrator
Default
Public
admin
peter.gibbons
3D Objects
AppData
Contacts
Desktop
Documents
Downloads
WebShell-master
Favorites
Links
Music
OneDrive
Pictures
Saved Games
Searches
Videos
Windows
D:
ntfs
registry
Name
Size
Mode
mtime
atime
ctime
btime
Customize.asmx.txt
0 Mb
-rw-rw-rw-
2022-07-12 00:55:43 UTC
2022-07-12 00:55:44 UTC
2022-07-12 00:55:43 UTC
2022-07-12 00:55:41 UTC
WebShell-master
0 b
drwxrwxrwx
2022-07-12 01:05:51 UTC
2023-01-11 04:13:40 UTC
2022-07-12 01:05:51 UTC
2022-07-12 01:05:51 UTC
WebShell-master.zip
25 Mb
-rw-rw-rw-
2022-07-12 01:05:40 UTC
2022-07-12 01:05:40 UTC
2022-07-12 01:05:40 UTC
2022-07-12 01:05:37 UTC
desktop.ini
0 Mb
-rw-rw-rw-
2022-07-01 06:11:30 UTC
2023-01-11 04:13:37 UTC
2022-07-01 06:11:30 UTC
2022-07-01 06:11:30 UTC
Stats
Textview
HexView
C:\Users\peter.gibbons\Downloads\WebShell-master.zip
Size
25870369
Mode
-rw-rw-rw-
Mtime
2022-07-12T01:05:40.2532364Z
Atime
2022-07-12T01:05:40.2532364Z
Ctime
2022-07-12T01:05:40.2532364Z
Btime
2022-07-12T01:05:37.6179076Z
Last Collected
2023-01-28 17:04:09 UTC
Fetch from Client
Re-Collect from the client
Properties
SHA256
47dd1ae6fab4c2cb108e2b139a4763ce5250b9f980b937d48552d985a1707535
MD5
e0480d4fc524853a2b13399080de8379
Launchpad
2023-01-28T17:04:23.323Z


Search clients
SAL-06.corp.net
Connected
admin
Microsoft.NET
Migration
ModemLogs
Offline Web Pages
PCHEALTH
PLA
Panther
Performance
PolicyDefinitions
Prefetch
Registration
Resources
SchCache
ServiceProfiles
Setup
ShellNew
SoftwareDistribution
Speech
SysWOW64
System32
TAPI
Tasks
Temp
20220919
20220920
CR_C3179.tmp
Crashpad
vmware-SYSTEM
{52D3D56A-49CF-4161-8126-1F618FB4
{6D1DAF48-67B3-44EF-BAD9-27C57D4
{A5A7A7E7-E0E6-4C27-8046-D6DA782C
{F437CC37-6A1F-45C2-9391-A193A4EF
Vss
Web
addins
assembly
debug
diagnostics
ehome
en-US
inf
rescache
schemas
security
Name
Size
Mode
mtime
atime
ctime
btime
PowerShell_transcript.SAL-06.6mt7gEl0.20220920081111.txt
0 Mb
-rw-rw-rw-
2022-09-20T13:11:12Z
2022-09-20T13:11:12Z
2022-09-20T13:11:12Z
2022-09-20T13:11:12Z
PowerShell_transcript.SAL-06.q_xqNPAc.20220920081133.txt
0 Mb
-rw-rw-rw-
2022-09-20T13:11:33Z
2022-09-20T13:11:33Z
2022-09-20T13:11:33Z
2022-09-20T13:11:33Z
Stats
Textview
HexView
1 ..**********..
2 Windows.PowerShell.transcript.start..
3 Start.time:.20220920081112..
4 Username:.CORP\SYSTEM..
5 RunAs.User:.CORP\SYSTEM..
6 Machine:.SAL-06.(Microsoft.Windows.NT.6.1.7601.Service.Pack.1)..
7 Host.Application:.Powershell.-NonInteractive.-NoProfile.msiexec.exe./i.GoogleChro
8 meStandaloneEnterprise64.msi./qn./norestart..
9 Process.ID:.3600..
10 PSVersion:.5.1.14409.1005..
11 PSEdition:.Desktop..
12 PSCompatibleVersions:.1.0,.2.0,.3.0,.4.0,.5.0,.5.1.14409.1005..
13 BuildVersion:.10.0.14409.1005..
14 CLRVersion:.4.0.30319.17929..
15 WSManStackVersion:.3.0..
16 PSRemotingProtocolVersion:.2.3..
17 SerializationVersion:.1.1.0.1..
18 **********..
19 **********..
20 Command.start.time:.20220920081112..
21 **********..
22 PS>msiexec.exe./i.GoogleChromeStandaloneEnterprise64.msi./qn./norestart..
23 **********..
24 Command.start.time:.20220920081112..
25 **********..
26 PS>$global:?..
27 True..
28 **********..
29 Windows.PowerShell.transcript.end..
30 End.time:.20220920081112..
31 **********..
32
2022-09-23T12:59:46Z


Search clients
SAL-06.corp.net
Connected
admin
auto
C:
D:
E:
ntfs
registry
HKEY_CLASSES_ROOT
HKEY_CURRENT_CONFIG
HKEY_CURRENT_USER
Control Panel
EUDC
Environment
Keyboard Layout
Printers
SYSTEM
Software
Classes
Microsoft
Mozilla
Policies
Sysinternals
System Monitor
HKEY_LOCAL_MACHINE
HKEY_PERFORMANCE_DATA
HKEY_USERS
Name
Size
Mode
mtime
atime
ctime
btime
EulaAccepted
0 Mb
-rwxr-xr-x
2022-09-19T23:31:32Z
2022-09-19T23:31:32Z
2022-09-19T23:31:32Z
2022-09-19T23:31:32Z
Stats
Textview
HexView
HKEY_CURRENT_USER\Software\Sysinternals\System Monitor\EulaAccepted
Size 4
Mode -rwxr-xr-x
Mtime 2022-09-19T23:31:32Z
Atime 2022-09-19T23:31:32Z
Ctime 2022-09-19T23:31:32Z
Btime 2022-09-19T23:31:32Z
Fetch from Client
Collect from the client
Properties
type DWORD
value 1
2022-09-23T13:01:03Z


Search clients
SAL-06.corp.net
Connected
admin
Interrogate
VFS
Collected
Overview
VQL Drilldown
Shell
SAL-06.corp.net
Client ID
C.121e71b72393dc41
Agent Version
2022-06-22T16:57:49+10:00
Agent Name
velociraptor
First Seen At
2022-09-20T13:03:18Z
Last Seen At
2022-09-23T12:52:23Z
Last Seen IP
70.123.40.66:52004
Labels
SAL
workstation
demo
Operating System
windows
Hostname
SAL-06
FQDN
SAL-06.corp.net
Release
Microsoft Windows 7 Professional Service Pack 16.1.7601 Build 7601
Architecture
amd64
Client Metadata
Key
Value
2022-09-23T12:52:25Z

New Hunt - Configure Hunt

Description: Get All Running Processes

Expiry: 9/30/2022 9:25 AM

Include Condition: Match by label

Include Labels: compromised ×

Exclude Condition: Run everywhere

Estimated affected clients 10

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Select artifacts to collect

pslist

Linux.Sys.Pslist

Windows.System.Pslist

Windows.System.Pslist

Type: client

List processes and their running binaries.

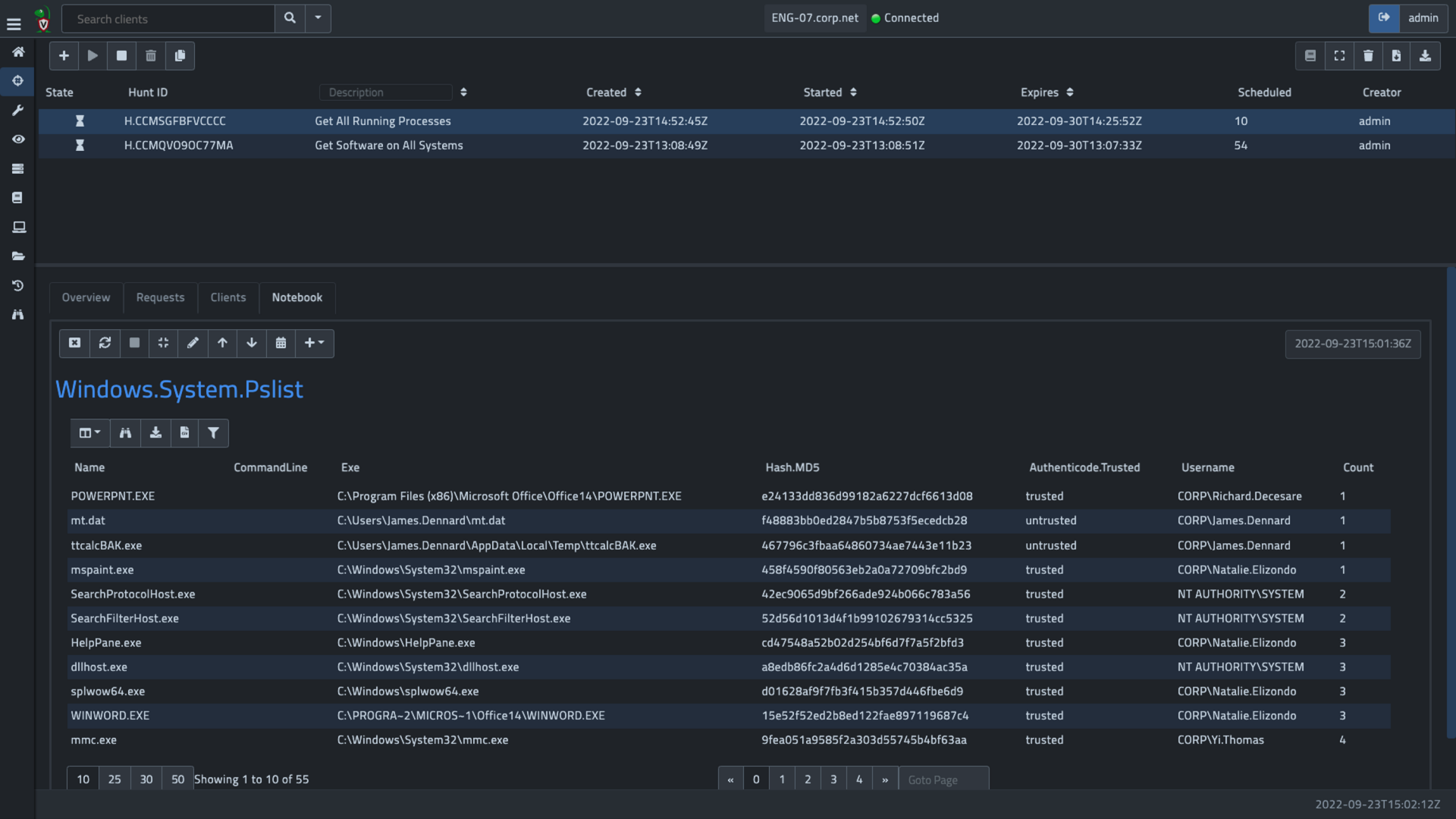
Parameters

Name	Type	Default	Description
processRegex	regex	.	

Source

```
SELECT Pid, Ppid, TokenIsElevated, Name, CommandLine, Exe,
       hash(path=Exe) as Hash,
       authenticode(filename=Exe) AS Authenticode,
       Username, Memory.WorkingSetSize AS WorkingSetSize
FROM pslist()
WHERE Name =~ processRegex

```

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch


Search clients
ENG-07.corp.net
Connected
admin
State
Hunt ID
Description
Created
Started
Expires
Scheduled
Creator
H.CCMSGFBFVCCCC
Get All Running Processes
2022-09-23T14:52:45Z
2022-09-23T14:52:50Z
2022-09-30T14:25:52Z
10
admin
H.CCMQVO9OC77MA
Get Software on All Systems
2022-09-23T13:08:49Z
2022-09-23T13:08:51Z
2022-09-30T13:07:33Z
54
admin
Overview
Requests
Clients
Notebook
2022-09-23T15:01:36Z
Windows.System.Pslist
Name
CommandLine
Exe
Hash.MD5
Authenticode.Trusted
Username
Count
POWERPNT.EXE
C:\Program Files (x86)\Microsoft Office\Office14\POWERPNT.EXE
e24133dd836d99182a6227dcf6613d08
trusted
CORP\Richard.Decesare
1
mt.dat
C:\Users\James.Dennard\mt.dat
f48883bb0ed2847b5b8753f5ecedcb28
untrusted
CORP\James.Dennard
1
ttcalcBAK.exe
C:\Users\James.Dennard\AppData\Local\Temp\ttcalcBAK.exe
467796c3fbaa64860734ae7443e11b23
untrusted
CORP\James.Dennard
1
mspaint.exe
C:\Windows\System32\mspaint.exe
458f4590f80563eb2a0a72709bfc2bd9
trusted
CORP\Natalie.Elizondo
1
SearchProtocolHost.exe
C:\Windows\System32\SearchProtocolHost.exe
42ec9065d9bf266ade924b066c783a56
trusted
NT AUTHORITY\SYSTEM
2
SearchFilterHost.exe
C:\Windows\System32\SearchFilterHost.exe
52d56d1013d4f1b99102679314cc5325
trusted
NT AUTHORITY\SYSTEM
2
HelpPane.exe
C:\Windows\HelpPane.exe
cd47548a52b02d254bf6d7f7a5f2bfd3
trusted
CORP\Natalie.Elizondo
3
dllhost.exe
C:\Windows\System32\dllhost.exe
a8edb86fc2a4d6d1285e4c70384ac35a
trusted
NT AUTHORITY\SYSTEM
3
splwow64.exe
C:\Windows\splwow64.exe
d01628af9f7fb3f415b357d446fbe6d9
trusted
CORP\Natalie.Elizondo
3
WINWORD.EXE
C:\PROGRA~2\MICROS~1\Office14\WINWORD.EXE
15e52f52ed2b8ed122fae897119687c4
trusted
CORP\Natalie.Elizondo
3
mmc.exe
C:\Windows\System32\mmc.exe
9fea051a9585f2a303d55745b4bf63aa
trusted
CORP\Yi.Thomas
4
10
25
30
50
Showing 1 to 10 of 55
«
0
1
2
3
4
»
Goto Page
2022-09-23T15:02:12Z

New Collection: Select Artifacts to collect

Windows.Sys.Programs

Windows.Sys.Programs

Windows.Sys.Programs

Type: client

Represents products as they are installed by Windows Installer. A product generally correlates to one installation package on Windows. Some fields may be blank as Windows installation details are left to the discretion of the product author.

Limitations: This key parses the live registry hives - if a user is not logged in then their data will not be resident in HKU and therefore you should parse the hives on disk (including within VSS/Regback).

Parameters

Name	Type	Default
programKeys		HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Unins HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentV HKEY_USERS*\Software\Microsoft\Windows\CurrentVersion\Uninstall*

Source

```
1
2  SELECT Key.Name as KeyName,
3         Key.Mtime AS KeyLastWriteTimestamp,
4         DisplayName,
5         DisplayVersion,
6         InstallLocation,
7         InstallSource,
8         Language,
9         Publisher,
10        UninstallString,
11        InstallDate,
12        Key.FullPath as KeyPath
13 FROM read_reg_key(globs=split(string=programKeys, sep=',[\\s]*'),
14                   accessor="registry")
```

Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Windows.Sys.Programs

KeyName	KeyLastWriteTimestamp	DisplayName	DisplayVersion	InstallLocation	InstallSource	Language	Publisher	UninstallString	InstallDate	KeyPath
Mozilla Firefox 68.0.1 (x64 en-US)	2022-09-20T13:12:50Z	Mozilla Firefox 68.0.1 (x64 en-US)	68.0.1	C:\Program Files\Mozilla Firefox			Mozilla	"C:\Program Files\Mozilla Firefox\uninstall\helper.exe"		HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\Mozilla Firefox 68.0.1 (x64 en-US)
MozillaMaintenanceService	2022-09-20T13:12:50Z	Mozilla Maintenance Service	68.0.1				Mozilla	"C:\Program Files (x86)\Mozilla Maintenance Service\uninstall.exe"		HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\MozillaMaintenanceService
{1AD147D0-BE0E-3D6C-AC11-64F6DC4163F1}	2022-09-19T19:33:00Z	Microsoft .NET Framework 4.5	4.5.50709		C:\42cc52b679e18619a1cfb5\	0	Microsoft Corporation	MsiExec.exe /X{1AD147D0-BE0E-3D6C-AC11-64F6DC4163F1}	20220919	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{1AD147D0-BE0E-3D6C-AC11-64F6DC4163F1}
{33620DCF-971A-4A0A-A217-D4178F006A2D}	2022-09-20T13:11:37Z	Velociraptor	0.65.3		C:\Windows\Temp\	1033	Velocidex	MsiExec.exe /X{33620DCF-971A-4A0A-A217-D4178F006A2D}	20220919	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{33620DCF-971A-4A0A-A217-D4178F006A2D}
{38624EB5-356D-4B08-8357-C33D89A5C0C5}	2022-09-19T23:32:37Z	Microsoft Visual C++ 2022 X64 Additional Runtime - 14.32.31326	14.32.31326		C:\ProgramData\Package Cache\{38624EB5-356D-4B08-8357-C33D89A5C0C5}v14.32.31326\packages\vcRuntimeAdditional_amd64\	1033	Microsoft Corporation	MsiExec.exe /I{38624EB5-356D-4B08-8357-C33D89A5C0C5}	20220919	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{38624EB5-356D-4B08-8357-C33D89A5C0C5}
{43D9111A-EA02-4682-BF3C-EFDCB62A89B0}	2022-09-19T19:42:06Z	VMware Tools	10.2.5.8068393	C:\Program Files\VMware\VMware Tools\	C:\Program Files\Common Files\VMware\InstallerCache\	1033	VMware, Inc.	MsiExec.exe /I{43D9111A-EA02-4682-BF3C-EFDCB62A89B0}	20220919	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{43D9111A-EA02-4682-BF3C-EFDCB62A89B0}
{50ADB1A8-7D22-3FA4-9F99-AD149455FE09}	2022-09-20T13:11:12Z	Google Chrome	105.0.5195.127		C:\Windows\Temp\	1033	Google LLC	MsiExec.exe /X{50ADB1A8-7D22-3FA4-9F99-AD149455FE09}	20220919	HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall\{50ADB1A8-7D22-3FA4-9F99-

New Hunt - Configure Hunt

Description: Get Software on All Systems

Expiry: 9/30/2022 8: 7 AM

Include Condition: Operating System

Operating System Included: Windows

Exclude Condition: Run everywhere

Estimated affected clients 55

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Select artifacts to collect

Windows.Sys.Programs

Windows.Sys.Programs

Windows.Sys.Programs

Type: client

Represents products as they are installed by Windows Installer. A product generally correlates to one installation package on Windows. Some fields may be blank as Windows installation details are left to the discretion of the product author.

Limitations: This key parses the live registry hives - if a user is not logged in then their data will not be resident in HKU and therefore you should parse the hives on disk (including within VSS/Regback).

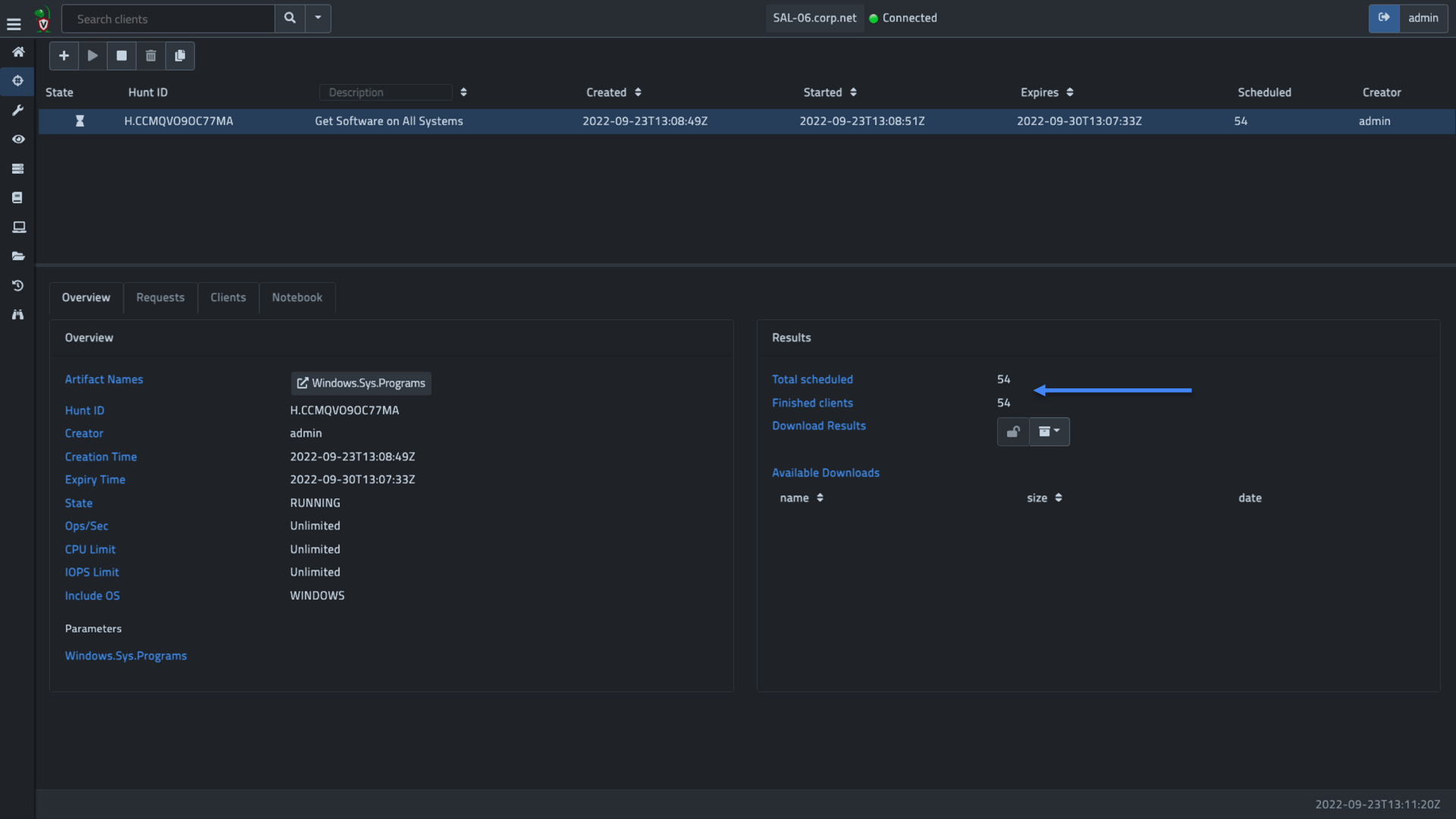
Parameters

Name	Type	Default
programKeys		HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninst HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Microsoft\Windows\CurrentV HKEY_USERS*\Software\Microsoft\Windows\CurrentVersion\Uninstall*

Source

```
1
2  SELECT Key.Name as KeyName,
3         Key.Mtime AS KeyLastWriteTimestamp,
4         DisplayName,
5         DisplayVersion,
6         InstallLocation,
7         InstallSource,
8         Language,
9         Publisher,
10        UninstallString,
11        InstallDate,
12        Key.FullPath as KeyPath
13 FROM read_reg_key(globs=split(string=programKeys, sep=',[\\s]*'),
14                   accessor="registry")
```

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch


Search clients
SAL-06.corp.net
Connected
admin
State
Hunt ID
Description
Created
Started
Expires
Scheduled
Creator
H.CCMQVO9OC77MA
Get Software on All Systems
2022-09-23T13:08:49Z
2022-09-23T13:08:51Z
2022-09-30T13:07:33Z
54
admin
Overview
Requests
Clients
Notebook
Overview
Artifact Names
Windows.Sys.Programs
Hunt ID
H.CCMQVO9OC77MA
Creator
admin
Creation Time
2022-09-23T13:08:49Z
Expiry Time
2022-09-30T13:07:33Z
State
RUNNING
Ops/Sec
Unlimited
CPU Limit
Unlimited
IOPS Limit
Unlimited
Include OS
WINDOWS
Parameters
Windows.Sys.Programs
Results
Total scheduled
54
Finished clients
54
Download Results
Available Downloads
name
size
date
2022-09-23T13:11:20Z

Overview | Requests | Clients | Notebook

VQL

```
/*
# Windows.Sys.Programs
*/
SELECT DisplayName,DisplayVersion,count() AS Count FROM source(artifact="Windows.Sys.Programs")
WHERE DisplayName
AND NOT DisplayName =~ "Language"
AND NOT DisplayName =~ "Microsoft"
GROUP BY DisplayName,DisplayVersion
ORDER BY Count
```

Windows.Sys.Programs

DisplayName	DisplayVersion	Count
Adobe Reader 9.2	9.2.0	1
Google Update Helper	1.3.35.451	1
Notepad++ (64-bit x64)	7.5.8	1
TightVNC	2.8.27.0	1
HMA! Pro VPN	4.6.151	1
Administative Templates for Windows 7 and Windows Server 2008 R2 (.admx)	1.0	1
CCleaner	5.33	1
Administrative Templates (ADMX) for Windows 10 Version 1511	1.0	1
Administrative Templates for Windows Server 2012 R2	1.0.0	1
Administrative Templates (ADMX) for Windows Server 2016 Technical Preview 5	1.0	1
WinRAR 5.50 beta 1 (32-bit)	5.50.1	1
CCleaner	6.04	1
VeraCrypt	1.22	1
Origin	10.5.115.51547	1
K-Lite v2.7		1
µTorrent	1.7.7	2

Real-Time Eventing & Log Forwarding

Event Monitoring: Select artifacts to collect from label group demo

Windows

- Windows.Detection.PsexecService
- Windows.Detection.PsexecService.Kill
- Windows.Detection.Service.Upload
- Windows.Detection.Thumbdrives.List
- Windows.Detection.Thumbdrives.OfficeKeywords
- Windows.Detection.Thumbdrives.OfficeMacros
- Windows.Detection.Usn
- Windows.Detection.WMIProcessCreation
- Windows.ETW.DNS
- Windows.ETW.DNSQueriesServer
- Windows.ETW.ETWSessions
- Windows.ETW.EdgeURLs
- Windows.ETW.Registry
- Windows.ETW.WMIProcessCreate
- Windows.Events.EventLogModifications
- Windows.Events.FailedLogBeforeSuccess
- Windows.Events.Kerbroasting

Windows.ETW.DNS

Type: client_event

Author: Matt Green - @mgreen27

This artifact monitors DNS queries using ETW.

There are several filteres availible to the user to filter out and target with regex, by default duplicate DNSCache requests are filtered out.

Parameters

Name	Type	Default	Description
ImageRegex	regex	.	ImagePath regex filter for
CommandLineRegex	regex	.	Commandline to filter for.
QueryRegex	regex	.	DNS query request (domain) to filter for.
AnswerRegex	regex	.	DNS answer to filter for.
CommandLineExclusion	regex	`svchost.exe -k NetworkService -p -s Dnscache$`	Commandline to filter out. Typically we do not want Dnscache events.

Source

```
1 LET RecentProcesses = SELECT * FROM fifo(query={
2           SELECT System.TimeStamp AS CreateTime,
3               EventData.ImageName AS ImageName,
```

Configure Label Group | Select Artifacts | Configure Parameters | Review | Launch

Event Monitoring: Select artifacts to collect from label group demo

- Windows.Detection.WMIProcessCreation
- Windows.ETW.DNS
- Windows.ETW.DNSQueriesServer
- Windows.ETW.ETWSessions
- Windows.ETW.EdgeURLs
- Windows.ETW.Registry
- Windows.ETW.WMIProcessCreate
- Windows.Events.EventLogModifications
- Windows.Events.FailedLogBeforeSuccess
- Windows.Events.Kerbroasting
- Windows.Events.ProcessCreation
- Windows.Events.ServiceCreation
- Windows.Events.TrackProcesses
- Windows.Events.TrackProcessesBasic
- Windows.Events.Trackaccount
- Windows.Forensics.LocalHashes.Usn
- Windows.Remediation.QuarantineMonitor
- Windows.Sysinternals.SysmonLogForward

Windows.Events.ProcessCreation

Type: client_event

Collect all process creation events.

Parameters

Name	Type	Default	Description
eventQuery		SELECT * FROM Win32_ProcessStartTrace	

Source

```
// Get information about the process
LET get_pid_query(Lpid) = SELECT Pid, Ppid, Name FROM if(condition=Lpid > 0,
then={
    SELECT Pid, Ppid, Name FROM pslist(pid=Lpid)
})

// Build the call chain. Cache the results for a short time.
LET pstree(LookupPid) = SELECT * FROM foreach(
  row=cache(func=get_pid_query(Lpid=LookupPid), key=str(str=LookupPid)),
  query={
      SELECT * FROM chain(
    a={
      SELECT Pid, Ppid, Name FROM scope()
    }, b={
      SELECT Pid, Ppid, Name FROM pstree(LookupPid=Ppid)
    })
})

LET call_chain(LookupPid) = SELECT Pid, Ppid, Name FROM
pstree(LookupPid=LookupPid)

```

Configure Label Group | Select Artifacts | Configure Parameters | Review | Launch

Event Monitoring: Select artifacts to collect from label group demo

- Windows.Detection.WMIProcessCreation
- Windows.ETW.DNS
- Windows.ETW.DNSQueriesServer
- Windows.ETW.ETWSessions
- Windows.ETW.EdgeURLs
- Windows.ETW.Registry
- Windows.ETW.WMIProcessCreate
- Windows.Events.EventLogModifications
- Windows.Events.FailedLogBeforeSuccess
- Windows.Events.Kerbroasting
- Windows.Events.ProcessCreation
- Windows.Events.ServiceCreation
- Windows.Events.TrackProcesses
- Windows.Events.TrackProcessesBasic
- Windows.Events.Trackaccount
- Windows.Forensics.LocalHashes.Usn
- Windows.Remediation.QuarantineMonitor
- Windows.Sysinternals.SysmonLogForward

Windows.Sysinternals.SysmonLogForward

Type: client_event

A client side event forwarder to forward sysmon events to the server.

Tools

- SysmonBinary
- SysmonConfig

Parameters

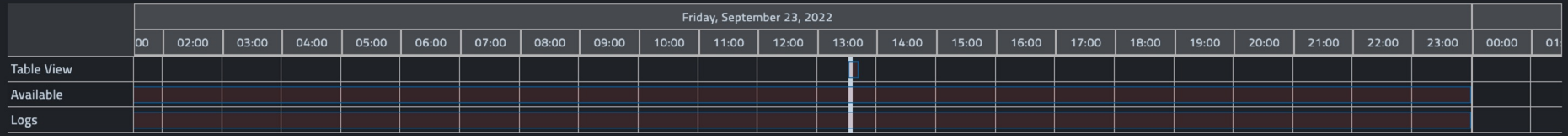
Name	Type	Default	Description
SysmonFileLocation		C:/Windows/sysmon64.exe	If set, we check this location first for sysmon installed.

Source

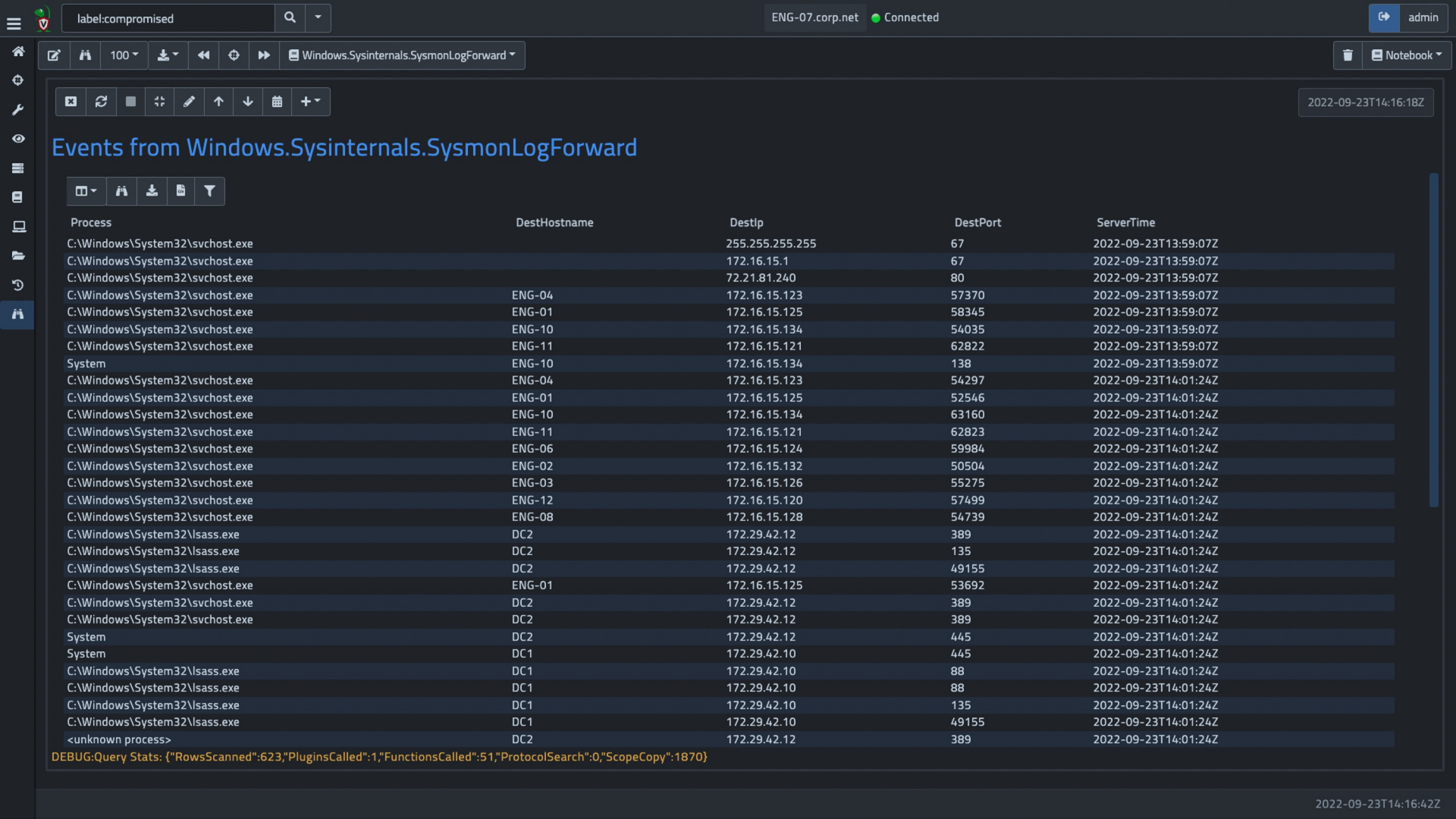
```
// First ensure that sysmon is actually installed.
LET _ <= SELECT * FROM Artifact.Windows.Sysinternals.SysmonInstall(
    SysmonFileLocation=SysmonFileLocation)

// Just parse and forward events. Use ETW rather than watch_evtx()
// because it is a little bit faster.
SELECT System.ID AS ID,
       System.TimeStamp AS Timestamp,
       get(member='EventData') AS EventData
FROM watch_etw(guid='{5770385f-c22a-43e0-bf4c-06f5698ffbd9}')

```

Configure Label Group | Select Artifacts | Configure Parameters | Review | Launch

100 Windows.Events.ProcessCreation Raw Data



Server Time	Timestamp	PPID	PID	Name	CommandLine	ParentInfo	CallChain	ClientId
2022-09-23T13:32:01Z	2022-09-23T13:29:12Z	508	4000	svchost.exe			svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:44Z	1796	2540	sysmon64.exe			sysmon64.exe <- Velociraptor.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:44Z	352	1716	conhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	1796	3572	sysmon64.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	352	2860	conhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	1796	3696	sysmon64.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	352	308	conhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	1796	3856	sysmon64.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	352	3952	conhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	1796	1164	sc.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:29:46Z	352	3672	conhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:00Z	912	3788	taskeng.exe			taskeng.exe <- svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:00Z	3788	300	gpupdate.exe			gpupdate.exe <- taskeng.exe <- svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:00Z	352	1248	conhost.exe			conhost.exe <- csrss.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:06Z	2180	3668	SearchProtocolHost.exe			SearchProtocolHost.exe <- SearchIndexer.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:06Z	2180	1700	SearchFilterHost.exe			SearchFilterHost.exe <- SearchIndexer.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:08Z	508	3732	taskhost.exe				C.121e71b72393dc41
2022-09-23T13:32:01Z	2022-09-23T13:30:24Z	508	432	raserver.exe				C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:39:54Z	912	1380	taskeng.exe			taskeng.exe <- svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:39:54Z	1380	2912	GoogleUpdate.exe			GoogleUpdate.exe <- taskeng.exe <- svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:00Z	1380	900	gpupdate.exe			gpupdate.exe <- taskeng.exe <- svchost.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:00Z	352	1688	conhost.exe			conhost.exe <- csrss.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:06Z	2180	3488	SearchProtocolHost.exe			SearchProtocolHost.exe <- SearchIndexer.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:06Z	2180	4028	SearchFilterHost.exe			SearchFilterHost.exe <- SearchIndexer.exe <- services.exe <- wininit.exe	C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:07Z	508	3088	taskhost.exe				C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:12Z	508	3136	GoogleUpdate.exe				C.121e71b72393dc41
2022-09-23T13:42:21Z	2022-09-23T13:40:23Z	508	3188	raserver.exe				C.121e71b72393dc41


label:compromised
ENG-07.corp.net
Connected
admin
100
Windows.Sysinternals.SysmonLogForward
Notebook
2022-09-23T14:16:18Z
Events from Windows.Sysinternals.SysmonLogForward
Process
DestHostname
DestIp
DestPort
ServerTime
C:\Windows\System32\svchost.exe 255.255.255.255 67 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe 172.16.15.1 67 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe 72.21.81.240 80 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe ENG-04 172.16.15.123 57370 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe ENG-01 172.16.15.125 58345 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe ENG-10 172.16.15.134 54035 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe ENG-11 172.16.15.121 62822 2022-09-23T13:59:07Z
System ENG-10 172.16.15.134 138 2022-09-23T13:59:07Z
C:\Windows\System32\svchost.exe ENG-04 172.16.15.123 54297 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-01 172.16.15.125 52546 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-10 172.16.15.134 63160 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-11 172.16.15.121 62823 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-06 172.16.15.124 59984 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-02 172.16.15.132 50504 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-03 172.16.15.126 55275 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-12 172.16.15.120 57499 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-08 172.16.15.128 54739 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC2 172.29.42.12 389 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC2 172.29.42.12 135 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC2 172.29.42.12 49155 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe ENG-01 172.16.15.125 53692 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe DC2 172.29.42.12 389 2022-09-23T14:01:24Z
C:\Windows\System32\svchost.exe DC2 172.29.42.12 389 2022-09-23T14:01:24Z
System DC2 172.29.42.12 445 2022-09-23T14:01:24Z
System DC1 172.29.42.10 445 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC1 172.29.42.10 88 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC1 172.29.42.10 88 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC1 172.29.42.10 135 2022-09-23T14:01:24Z
C:\Windows\System32\lsass.exe DC1 172.29.42.10 49155 2022-09-23T14:01:24Z
<unknown process> DC2 172.29.42.12 389 2022-09-23T14:01:24Z
DEBUG:Query Stats: {"RowsScanned":623,"PluginsCalled":1,"FunctionsCalled":51,"ProtocolSearch":0,"ScopeCopy":1870}
2022-09-23T14:16:42Z

Threat Hunting & Detection

Event Monitoring: Select artifacts to collect from label group demo

- Windows.Detection.WMIProcessCreation
- Windows.ETW.DNS
- Windows.ETW.DNSQueriesServer
- Windows.ETW.ETWSessions
- Windows.ETW.EdgeURLs
- Windows.ETW.Registry
- Windows.ETW.WMIProcessCreate
- Windows.Events.EventLogModifications
- Windows.Events.FailedLogBeforeSuccess
- Windows.Events.Kerbroasting
- Windows.Events.ProcessCreation
- Windows.Events.ServiceCreation
- Windows.Events.TrackProcesses
- Windows.Events.TrackProcessesBasic
- Windows.Events.Trackaccount
- Windows.Forensics.LocalHashes.Usn
- Windows.Remediation.QuarantineMonitor
- Windows.Sysinternals.SysmonLogForward

Windows.Events.Kerbroasting

Type: client_event

Author: Matt Green - @mgreen27

Description: This Artifact will monitor all successful Kerberos TGS Ticket events for Service Accounts (SPN attribute) implemented with weak encryption. These tickets are vulnerable to brute force attack and this event is an indicator of a Kerbroasting attack.

ATT&CK: T1208 - Kerbroasting Typical attacker methodology is to firstly request accounts in the domain with SPN attributes, then request an insecure TGS ticket for brute forcing. This attack is particularly effective as any domain credentials can be used to implement the attack and service accounts often have elevated privileges. Kerbroasting can be used for privilege escalation or persistence by adding a SPN attribute to an unexpected account.

Reference: The Art of Detecting Kerberoast Attacks Log Source: Windows Security Event Log (Domain Controllers) Event ID: 4769 Status: 0x0 (Audit Success) Ticket Encryption: 0x17 (RC4) Service Name: NOT krbtgt or NOT a system account (account name ends in $) TargetUserName: NOT a system account (*$@*)

Monitor and alert on unusual events from an unexpected IP. Note: There are potential false positives so whitelist normal source IPs and manage risk of insecure ticket generation.

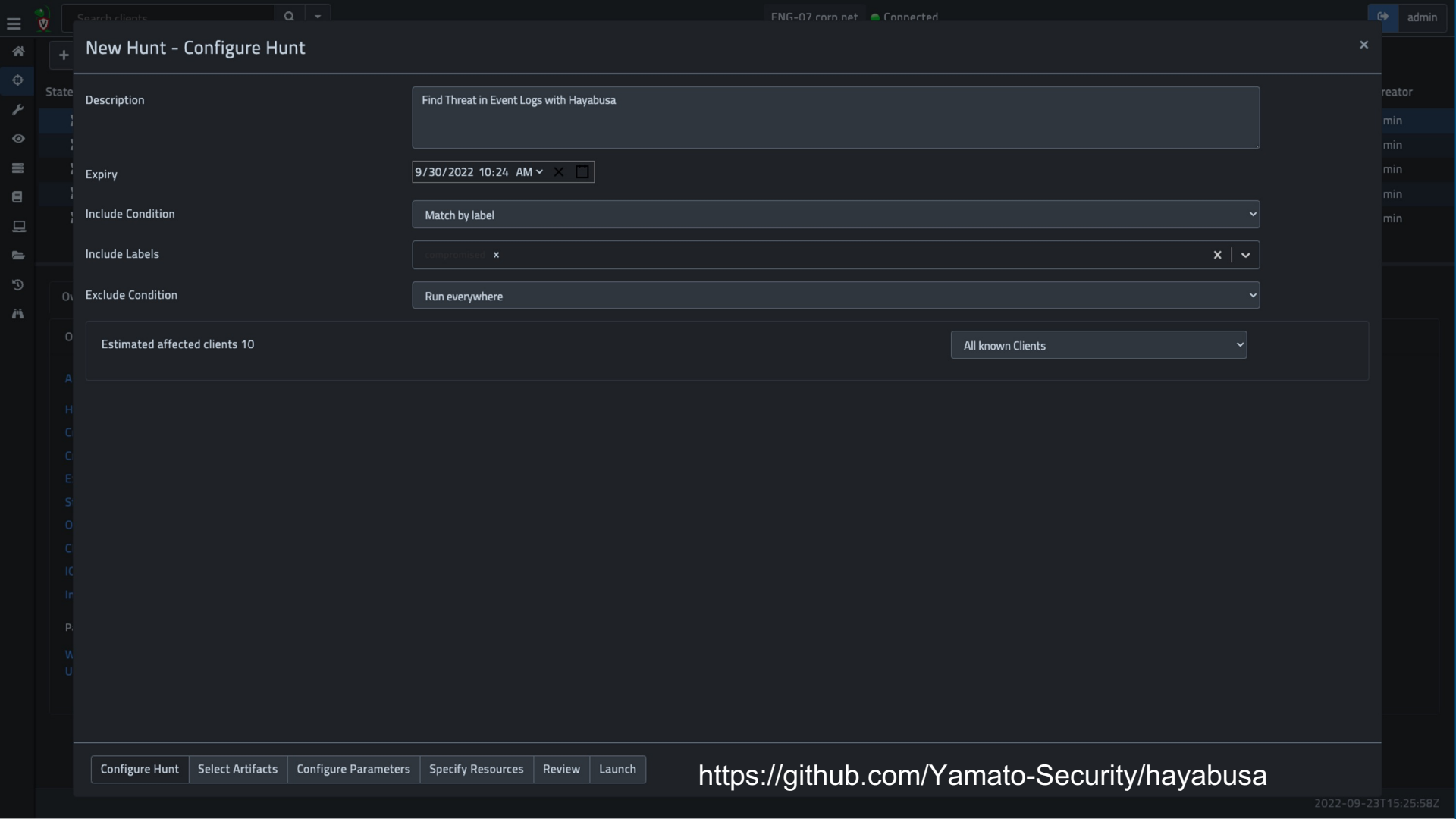
Parameters

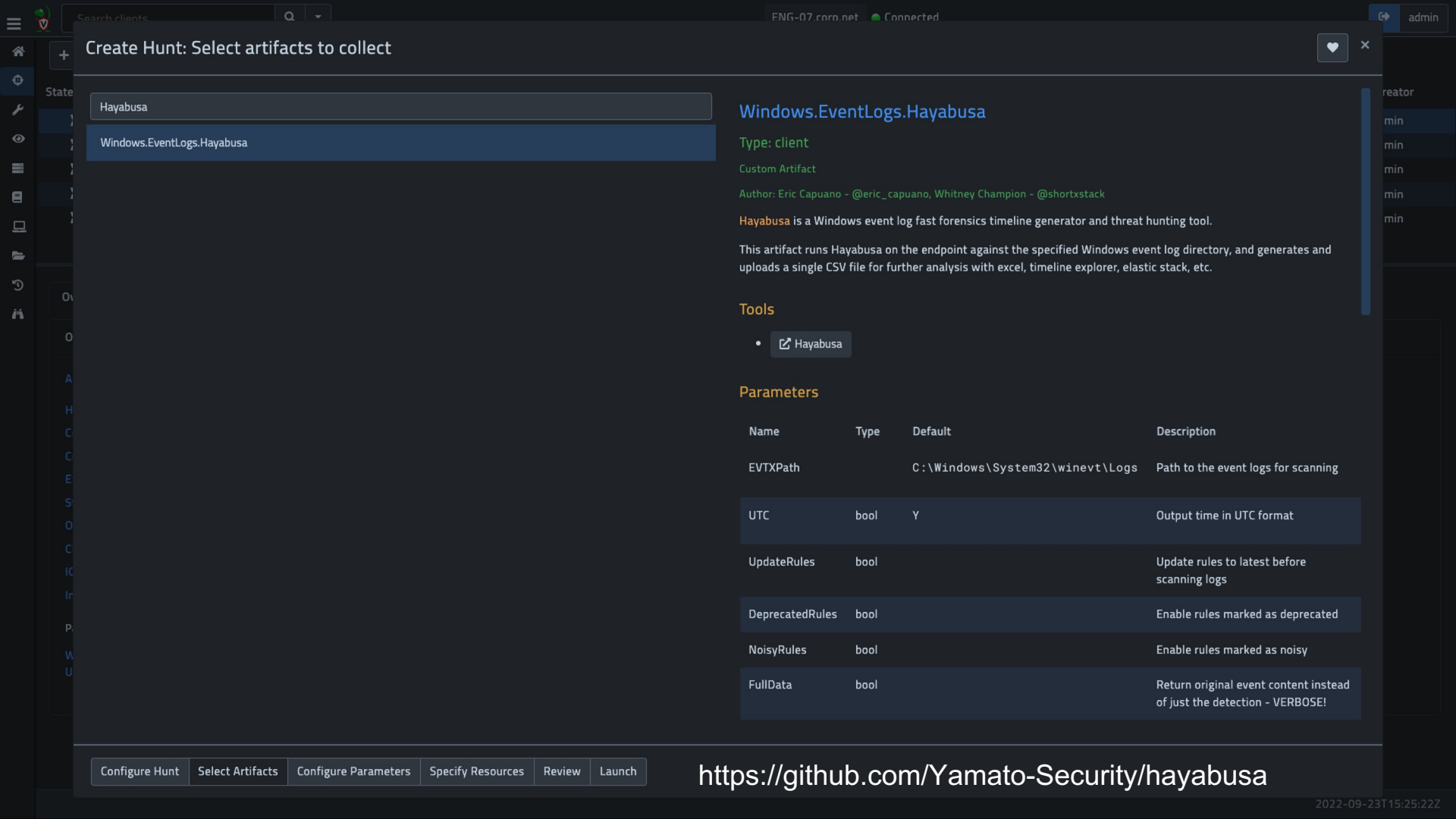
Name	Type	Default	Description
eventLog		C:\Windows\system32\winevt\logs\Security.evtx	

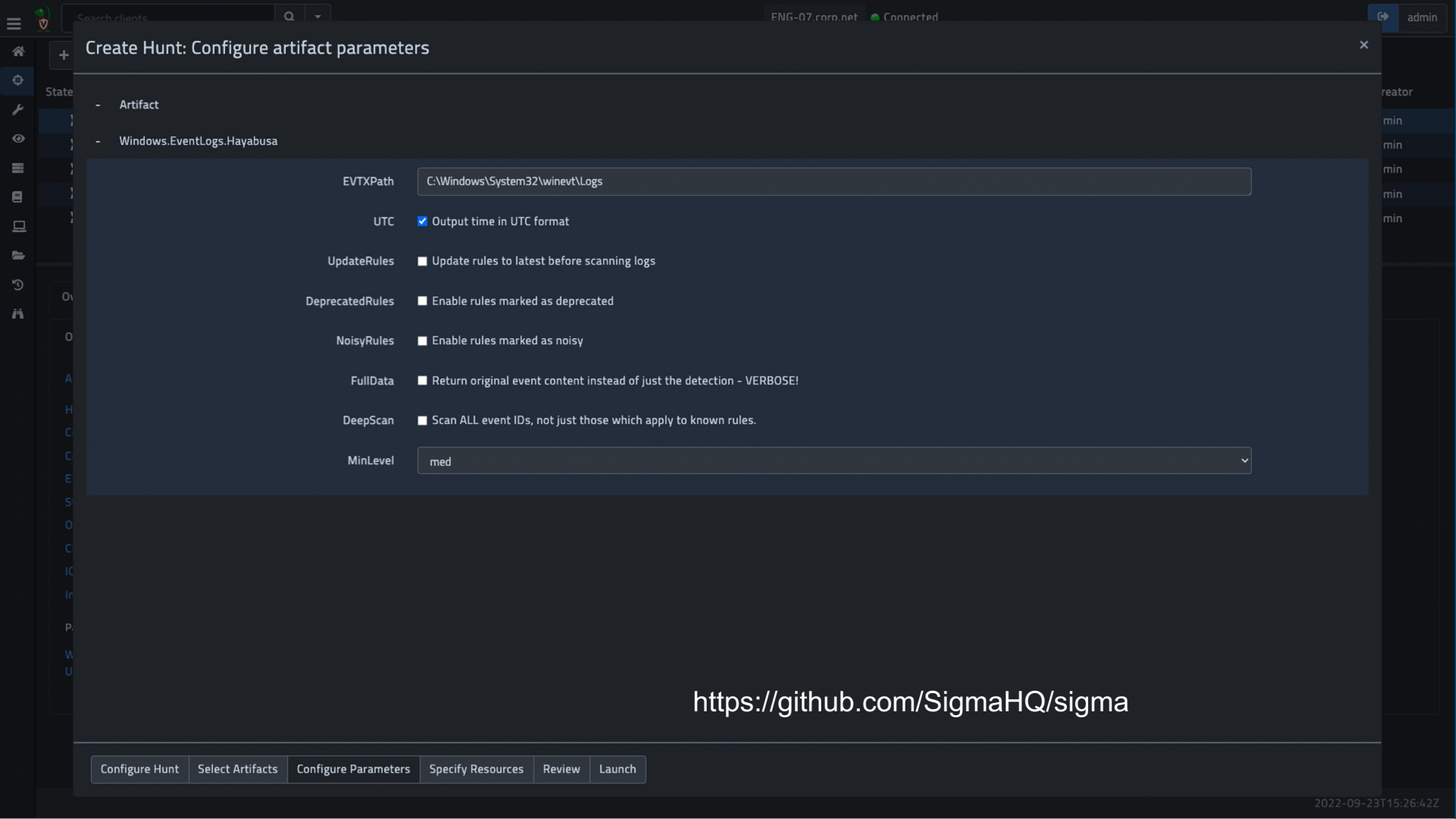
Source Kerbroasting

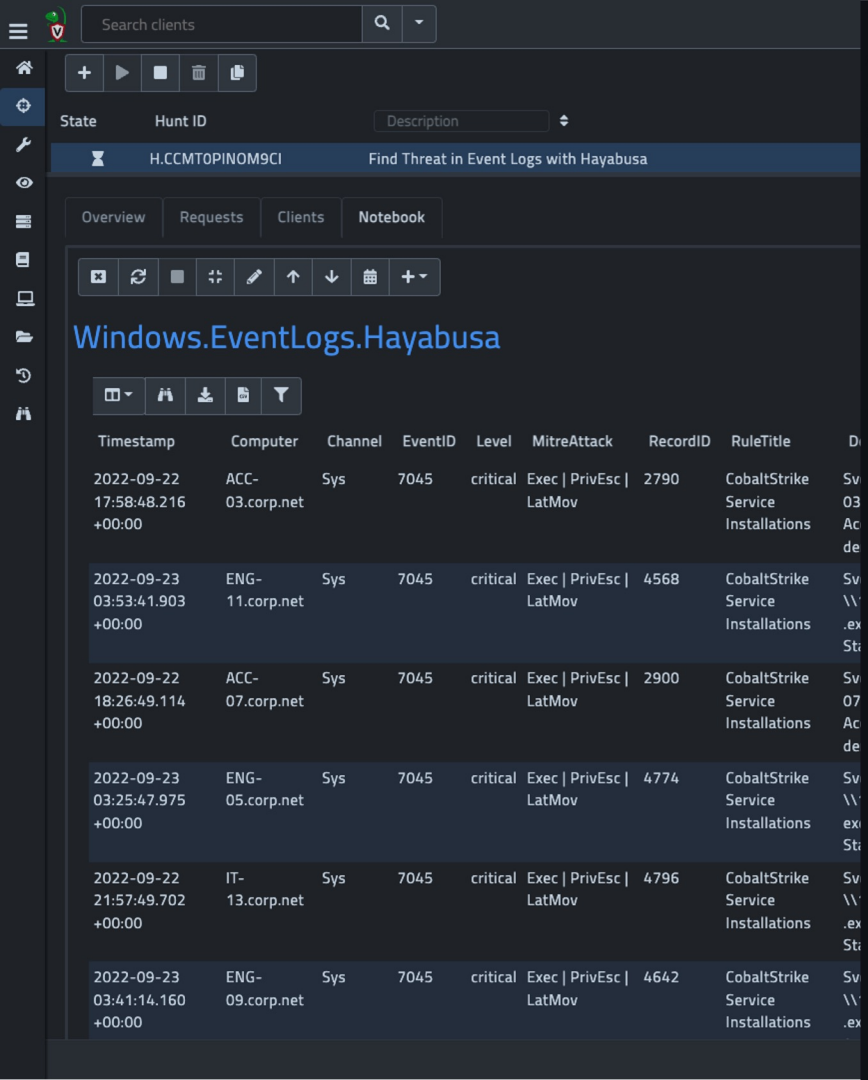
```
1 LET files = SELECT * FROM glob(globs=eventLog)
2
3 SELECT timestamp(epoch=System.TimeCreated.SystemTime) As EventTime,
4        System.EventID.Value as EventID,
5        System.Computer as Computer,
6        EventData.ServiceName as ServiceName,
```

Configure Label Group | Select Artifacts | Configure Parameters | Review | Launch


New Hunt - Configure Hunt
Description
Find Threat in Event Logs with Hayabusa
Expiry
9/30/2022 10:24 AM
Include Condition
Match by label
Include Labels
Exclude Condition
Run everywhere
Estimated affected clients 10
All known Clients
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch
https://github.com/Yamato-Security/hayabusa


Create Hunt: Select artifacts to collect
Hayabusa
Windows.EventLogs.Hayabusa
Windows.EventLogs.Hayabusa
Type: client
Custom Artifact
Author: Eric Capuano - @eric_capuano, Whitney Champion - @shortxstack
Hayabusa is a Windows event log fast forensics timeline generator and threat hunting tool.
This artifact runs Hayabusa on the endpoint against the specified Windows event log directory, and generates and uploads a single CSV file for further analysis with excel, timeline explorer, elastic stack, etc.
Tools
Hayabusa
Parameters
Name Type Default Description
EVTXPath C:\Windows\System32\winevt\Logs Path to the event logs for scanning
UTC bool Y Output time in UTC format
UpdateRules bool Update rules to latest before scanning logs
DeprecatedRules bool Enable rules marked as deprecated
NoisyRules bool Enable rules marked as noisy
FullData bool Return original event content instead of just the detection - VERBOSE!
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch
https://github.com/Yamato-Security/hayabusa


Create Hunt: Configure artifact parameters
- Artifact
- Windows.EventLogs.Hayabusa
EVTXPath
C:\Windows\System32\winevt\Logs
UTC
Output time in UTC format
UpdateRules
Update rules to latest before scanning logs
DeprecatedRules
Enable rules marked as deprecated
NoisyRules
Enable rules marked as noisy
FullData
Return original event content instead of just the detection - VERBOSE!
DeepScan
Scan ALL event IDs, not just those which apply to known rules.
MinLevel
med
https://github.com/SigmaHQ/sigma
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch




```yaml
title: Meterpreter or Cobalt Strike Getsystem Service Installation
id: 843544a7-56e0-4dcc-a44f-5cc266dd97d6
description: Detects the use of getsystem Meterpreter/Cobalt Strike command by detecting a specific service installation
status: experimental
author: Teymur Kheirkhabarov, Ecco, Florian Roth
date: 2019/10/26
modified: 2022/02/01
references:
    - https://speakerdeck.com/heirhabarov/hunting-for-privilege-escalation-in-windows-environment
    - https://blog.cobaltstrike.com/2014/04/02/what-happens-when-i-type-getsystem/
tags:
    - attack.privilege_escalation
    - attack.t1134.001
    - attack.t1134.002
logsource:
    product: windows
    service: system
detection:
    selection_id:
        Provider_Name: 'Service Control Manager'
        EventID: 7045
    selection:
        # meterpreter getsystem technique 1: cmd.exe /c echo 559891bb017 > \\.\pipe\5e120a
        - ImagePath|contains|all:
            - 'cmd'
            - '/c'
            - 'echo'
            - '\pipe\'
        # cobaltstrike getsystem technique 1: %COMSPEC% /c echo 559891bb017 > \\.\pipe\5e120a
        - ImagePath|contains|all:
            - '%COMSPEC%'
            - '/c'
            - 'echo'
            - '\pipe\'
        # cobaltstrike getsystem technique 1b (expanded %COMSPEC%): %COMSPEC% /c echo 559891bb017 > \\.\pipe\5e120a
        - ImagePath|contains|all:
            - 'cmd.exe'
            - '/c'
            - 'echo'
            - '\pipe\'
        # meterpreter getsystem technique 2: rundll32.exe C:\Users\test\AppData\Local\Temp\tmexsn.dll,a /p:tmexsn
        - ImagePath|contains|all:
            - 'rundll32'
            - '.dll,a'
            - '/p:'
        - ImagePath|startswith: '\\\\127.0.0.1\\ADMIN$\'  # https://twitter.com/svch0st/status/1413688851877416960?lang=en
```

https://github.com/SigmaHQ/sigma

New Hunt - Configure Hunt

Description: Yara Scan Processes

Expiry: 9/30/2022 10: 3 AM

Include Condition: Operating System

Operating System Included: Windows

Exclude Condition: Run everywhere

Estimated affected clients 55

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Select artifacts to collect

Detection.Yara.Process

Linux.Detection.Yara.Process

Windows.Detection.Yara.Process

Windows.Detection.Yara.Process

Type: client

Author: Matt Green - @mgreen27

This artifact enables running Yara over processes in memory.

There are 3 kinds of Yara rules that can be deployed: 1. Url link to a yara rule. 2. Shorthand yara in the format "wide nocase ascii:string1,string2,string3". 3. or a Standard Yara rule attached as a parameter. Only one method of Yara will be applied and search order is as above. The default is Cobalt Strike opcodes.

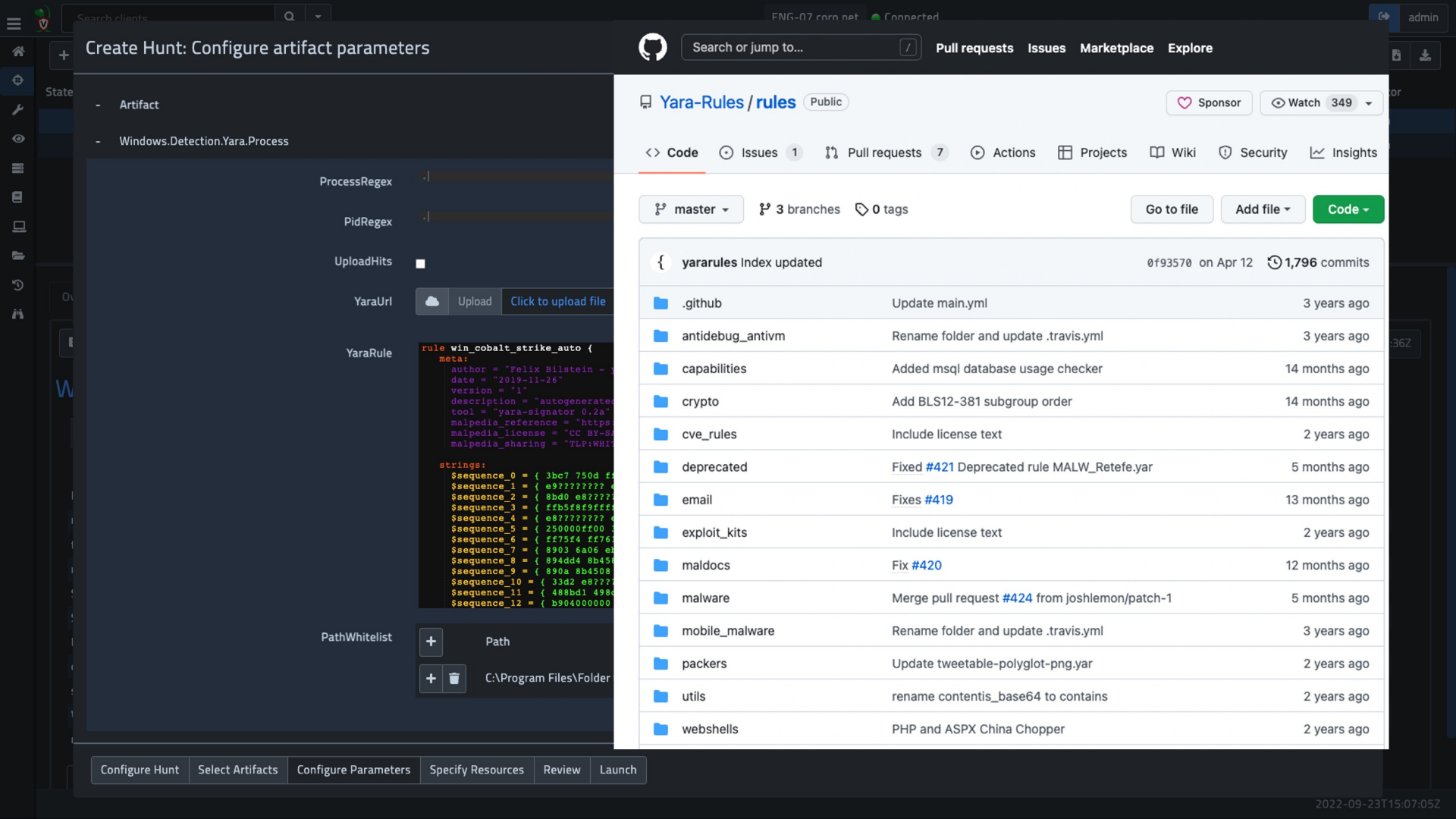
Regex parameters can be applied for process name and pid for targeting. The artifact also has an option to upload any process with Yara hits.

Note: the Yara scan will stop after one hit. Multi-string rules will also only show one string in returned rows.

Parameters

Name	Type	Default
ProcessRegex	regex	.
PidRegex	regex	.
UploadHits	bool	
YaraUrl	upload	
YaraRule	yara	rule win_cobalt_strike_auto {

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch


Create Hunt: Configure artifact parameters
Artifact
Windows.Detection.Yara.Process
ProcessRegex
PidRegex
UploadHits
YaraUrl
Upload
Click to upload file
YaraRule
rule win_cobalt_strike_auto {
meta:
strings:
PathWhitelist
Path
C:\Program Files\Folder
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch
Search or jump to...
Pull requests
Issues
Marketplace
Explore
Yara-Rules / rules
Public
Sponsor
Watch 349
Code
Issues 1
Pull requests 7
Actions
Projects
Wiki
Security
Insights
master
3 branches
0 tags
Go to file
Add file
Code
yararules Index updated
0f93570 on Apr 12
1,796 commits
.github Update main.yml 3 years ago
antidebug_antivm Rename folder and update .travis.yml 3 years ago
capabilities Added msql database usage checker 14 months ago
crypto Add BLS12-381 subgroup order 14 months ago
cve_rules Include license text 2 years ago
deprecated Fixed #421 Deprecated rule MALW_Retefe.yar 5 months ago
email Fixes #419 13 months ago
exploit_kits Include license text 2 years ago
maldocs Fix #420 12 months ago
malware Merge pull request #424 from joshlemon/patch-1 5 months ago
mobile_malware Rename folder and update .travis.yml 3 years ago
packers Update tweetable-polyglot-png.yar 2 years ago
utils rename contentis_base64 to contains 2 years ago
webshells PHP and ASPX China Chopper 2 years ago

New Hunt - Configure Hunt

Description: Find IOCs across systems

Expiry: 9/30/2022 10:16 AM

Include Condition: Operating System

Operating System Included: Windows

Exclude Condition: Run everywhere

Estimated affected clients 55

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Configure artifact parameters

- Artifact
- Windows.Search.FileFinder

SearchFilesGlob	C:\Users*
SearchFilesGlobTable	Glob C:**\ttcalcBAK.exe C:**\mt.dat
Accessor	auto
YaraRule	
Upload_File	☐
Calculate_Hash	☐
MoreRecentThan	--/--/---- --:-- -- UTC
ModifiedBefore	--/--/---- --:-- -- UTC

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

State	Hunt ID	Description	Created	Started	Expires	Scheduled	Creator
⧗	H.CCMSSG2NOD06I	Find IOCs across systems	2022-09-23T15:18:24Z	2022-09-23T15:18:28Z	2022-09-30T15:17:35Z	54	admin
⧗	H.CCMSNHHIRG192	Yara Scan Processes	2022-09-23T15:07:50Z	2022-09-23T15:07:54Z	2022-09-30T15:03:32Z	54	admin
⧗	H.CCMSGFBFVCCCC	Get All Running Processes	2022-09-23T14:52:45Z	2022-09-23T14:52:50Z	2022-09-30T14:25:52Z	10	admin
⧗	H.CCMQVO9OC77MA	Get Software on All Systems	2022-09-23T13:08:49Z	2022-09-23T13:08:51Z	2022-09-30T13:07:33Z	54	admin

Overview | Requests | Clients | Notebook

2022-09-23T15:20:04Z

Windows.Search.FileFinder

FullPath	Inode	Mode	Size	MTime	ATime	CTime	BTime	Keywords	IsDir	Upload	Hash	Data	FlowId	ClientId	Fqdn
C:\Users\James.Dennard\mt.dat		-rw-rw-rw-	2321408	2022-09-22T21:49:58Z	2022-09-22T21:48:03Z	2022-09-22T21:49:58Z	2022-09-22T21:48:03Z		false			{ }	F.CCMSSHK2O6JE4	C.9324449bf6703d33	IT-13.corp.net
C:\Users\James.Dennard\AppData\Local\Temp\ttcalcBAK.exe		-rw-rw-rw-	14336	2022-09-22T21:52:14Z	2022-09-22T21:52:14Z	2022-09-22T21:52:14Z	2022-09-22T21:52:14Z		false			{ }	F.CCMSSHK2O6JE4	C.9324449bf6703d33	IT-13.corp.net

10 | 25 | 30 | 50 Showing 1 to 2 of 2

« 0 » Goto Page

DEBUG:Query Stats: {"RowsScanned":2,"PluginsCalled":1,"FunctionsCalled":0,"ProtocolSearch":0,"ScopeCopy":5}

Incident Response, Containment, & Remediation

New Hunt - Configure Hunt

Description: Find every system compromised user has logged onto

Expiry: 9/30/2022 10:20 AM

Include Condition: Operating System

Operating System Included: Windows

Exclude Condition: Run everywhere

Estimated affected clients 55

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Select artifacts to collect

rdp

Windows.EventLogs.RDPAuth

Windows.EventLogs.RDPAuth

Type: client

Author: Matt Green - @mgreen27

This artifact will extract Event Logs related to Remote Desktop sessions, logon and logoff.

Security channel - EventID in 4624,4634 AND LogonType 3, 7, or 10. Security channel - EventID in 4778,4625,4779, or 4647. System channel - EventID 9009. Microsoft-Windows-TerminalServices-RemoteConnectionManager/Operational - EventID 1149. Microsoft-Windows-TerminalServices-LocalSessionManager/Operational - EventID 23,22,21,24,25,39, or 40.

Best use of this artifact is to collect RDP and Authentication events around a timeframe of interest and order by EventTime to scope RDP activity.

Parameters

Name	Type	Default	De
Security		%SystemRoot%\System32\Winevt\Logs\Security.evtx	pat
System		%SystemRoot%\System32\Winevt\Logs\System.evtx	pat
LocalSessionManager		%SystemRoot%\System32\Winevt\Logs\Microsoft-Windows-TerminalServices-LocalSessionManager%4Operational.evtx	pat Loc op
RemoteConnectionManager		%SystemRoot%\System32\Winevt\Logs\Microsoft-Windows-TerminalServices-RemoteConnectionManager%4Operational.evtx	pat Re op
DateAfter	timestamp		sea dat

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Create Hunt: Configure artifact parameters

- Artifact
- Windows.EventLogs.RDPAuth

Security	%SystemRoot%\System32\Winevt\Logs\Security.evtx
System	%SystemRoot%\System32\Winevt\Logs\System.evtx
LocalSessionManager	%SystemRoot%\System32\Winevt\Logs\Microsoft-Windows-TerminalServices-LocalSessionManager%4Operational.evtx
RemoteConnectionManager	%SystemRoot%\System32\Winevt\Logs\Microsoft-Windows-TerminalServices-RemoteConnectionManager%4Operational.evtx
DateAfter	--/--/---- --:-- -- UTC
DateBefore	--/--/---- --:-- -- UTC
SourceIPRegex	.
UserNameRegex	James.Dennard
UserNameWhitelist	\$$
SearchVSS	add VSS into query.

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

Overview | Requests | Clients | Notebook

2022-09-23T15:32:28Z

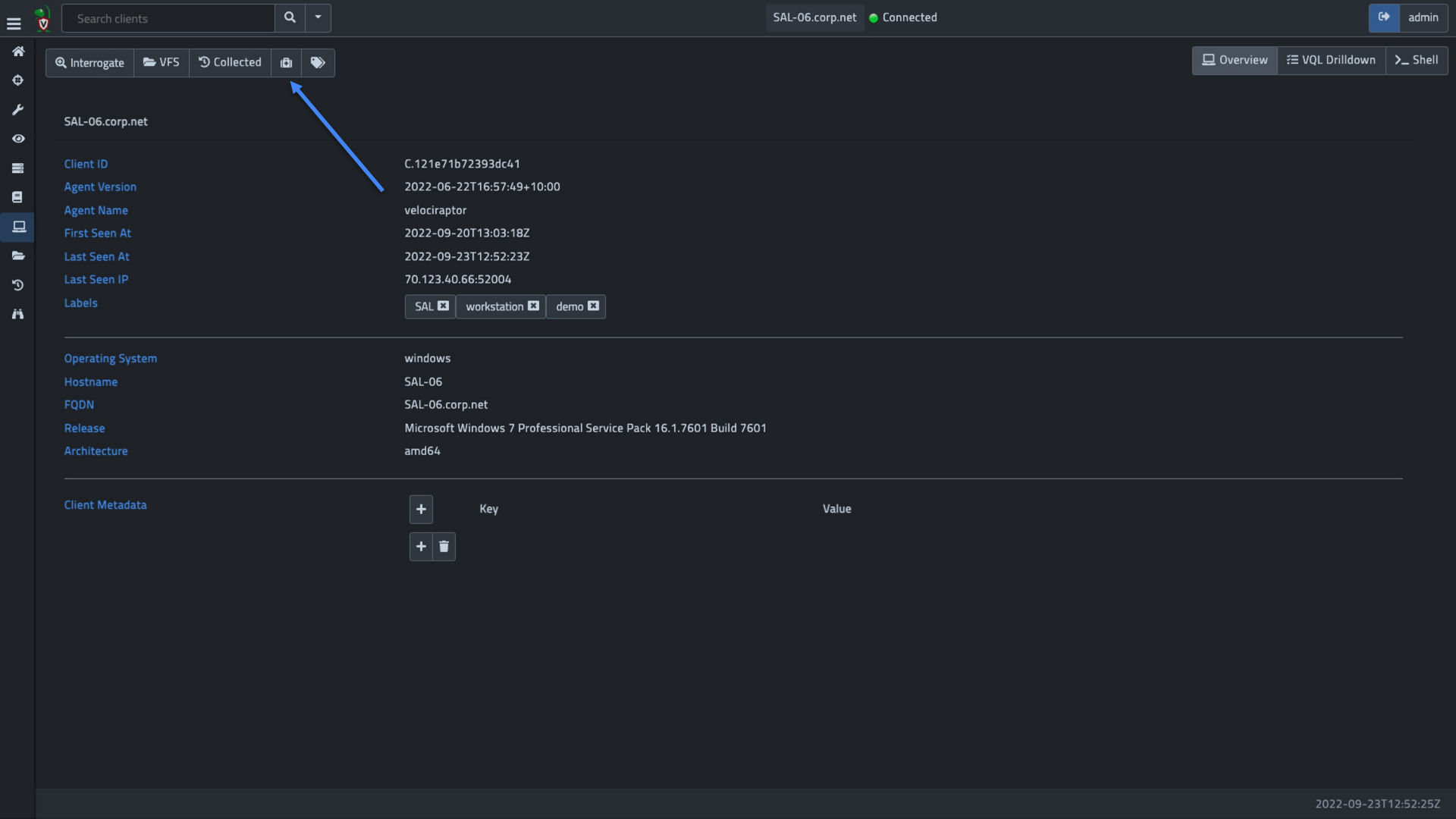
Windows.EventLogs.RDPAuth

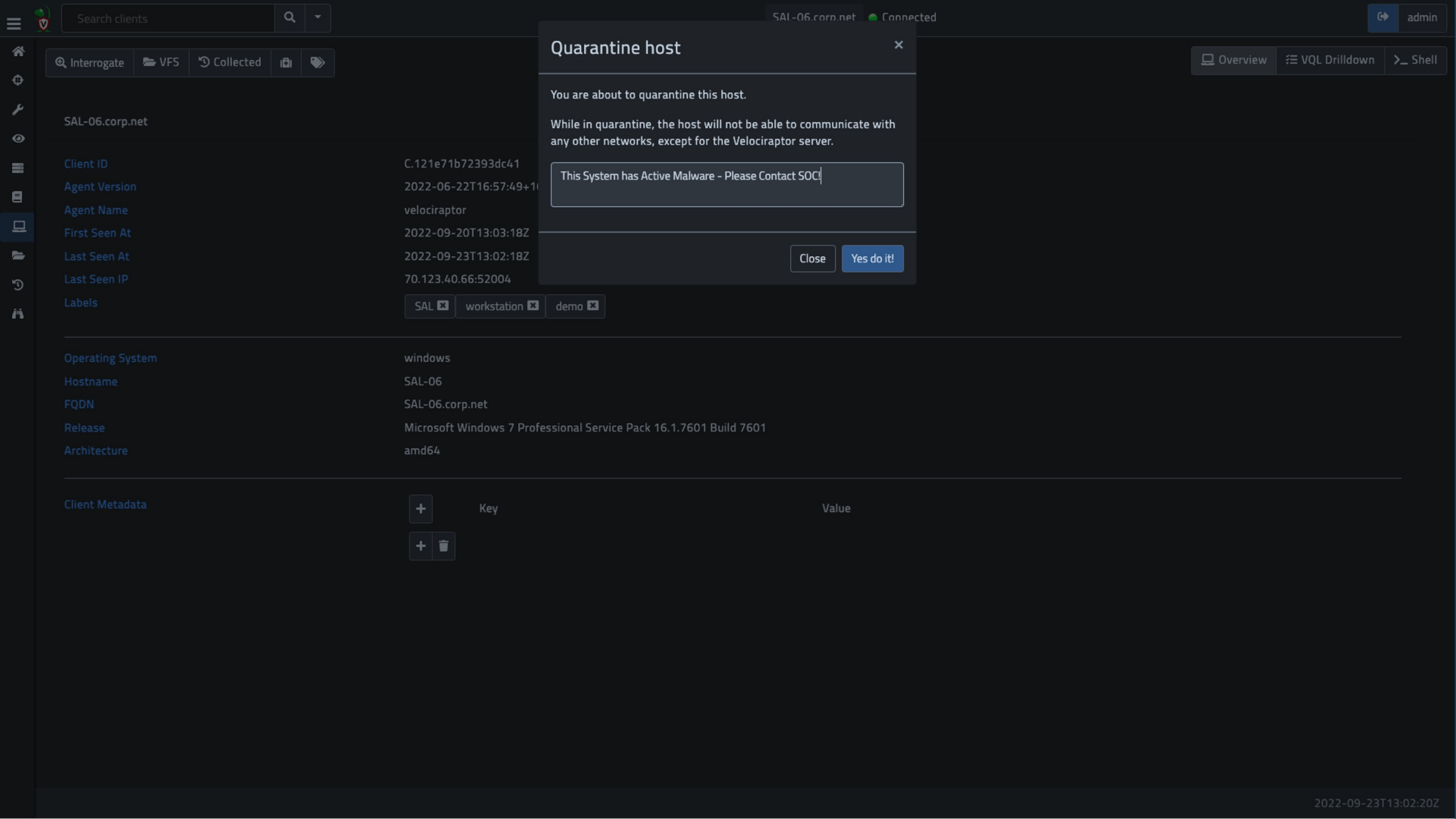
Computer	EventID	UserName	LogonType	SourceIP	Description	Count
ENG-01.corp.net	4624	James.Dennard	3	-	LOGON_SUCCESSFUL	9
IT-13.corp.net	4624	James.Dennard	7	-	LOGON_SUCCESSFUL_OLD	6
ENG-01.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	3
ENG-02.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
ENG-04.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
ENG-06.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
IT-08.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
ENG-09.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
IT-11.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
ENG-03.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1
ENG-11.corp.net	4624	James.Dennard	3	172.16.17.223	LOGON_SUCCESSFUL	1

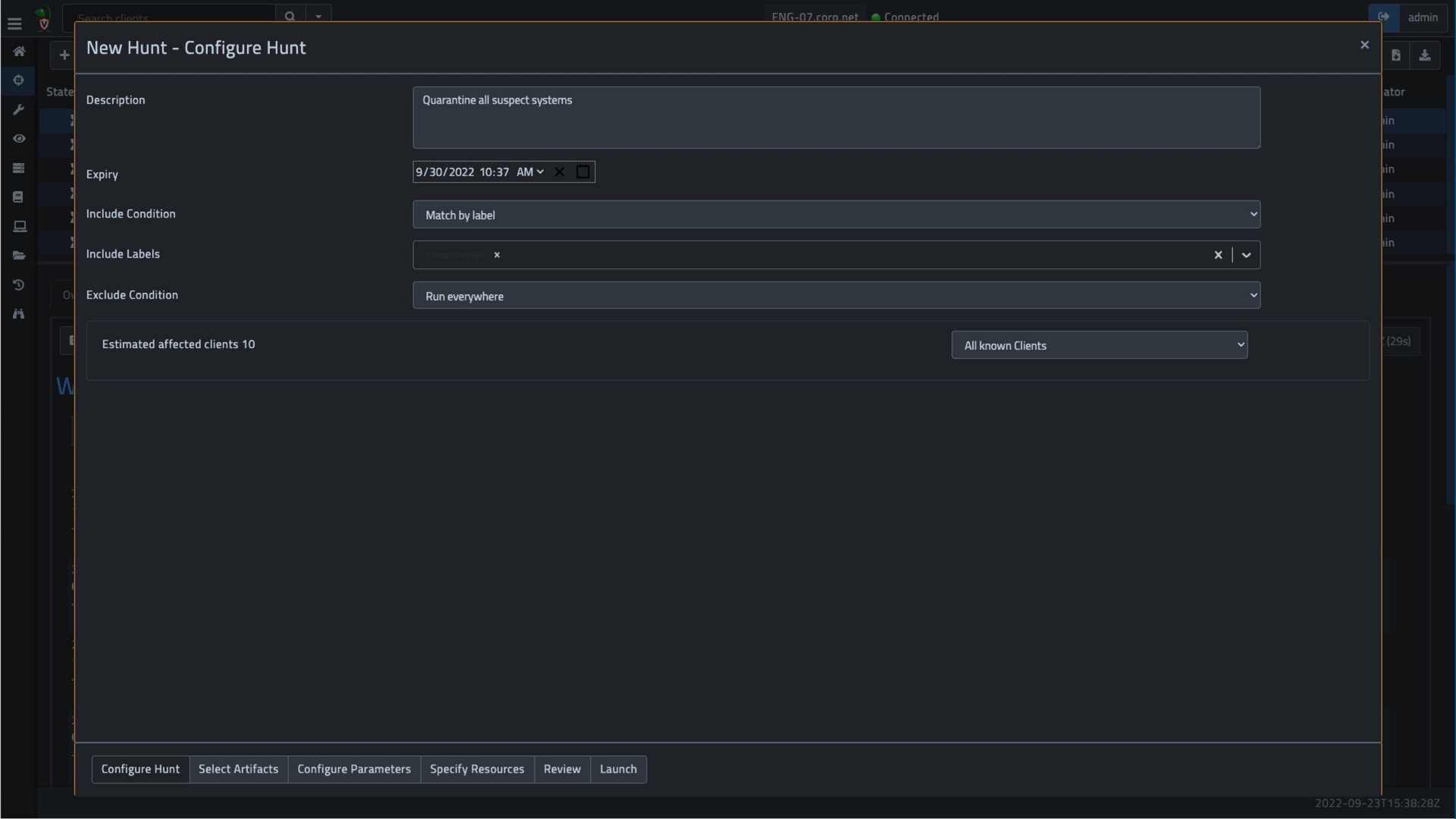
10 | 25 | 30 | 50 Showing 1 to 10 of 26

« 0 1 2 » Goto Page

DEBUG:Query Stats: {"RowsScanned":5389,"PluginsCalled":1,"FunctionsCalled":41,"ProtocolSearch":0,"ScopeCopy":10779}


Search clients
SAL-06.corp.net
Connected
admin
Interrogate
VFS
Collected
Overview
VQL Drilldown
Shell
SAL-06.corp.net
Client ID
C.121e71b72393dc41
Agent Version
2022-06-22T16:57:49+10:00
Agent Name
velociraptor
First Seen At
2022-09-20T13:03:18Z
Last Seen At
2022-09-23T12:52:23Z
Last Seen IP
70.123.40.66:52004
Labels
SAL
workstation
demo
Operating System
windows
Hostname
SAL-06
FQDN
SAL-06.corp.net
Release
Microsoft Windows 7 Professional Service Pack 16.1.7601 Build 7601
Architecture
amd64
Client Metadata
Key
Value
2022-09-23T12:52:25Z


Search clients
SAL-06.corp.net
Connected
admin
Interrogate
VFS
Collected
Overview
VQL Drilldown
Shell
Quarantine host
You are about to quarantine this host.
While in quarantine, the host will not be able to communicate with any other networks, except for the Velociraptor server.
This System has Active Malware - Please Contact SOC!
Close
Yes do it!
SAL-06.corp.net
Client ID
C.121e71b72393dc41
Agent Version
2022-06-22T16:57:49+1
Agent Name
velociraptor
First Seen At
2022-09-20T13:03:18Z
Last Seen At
2022-09-23T13:02:18Z
Last Seen IP
70.123.40.66:52004
Labels
SAL
workstation
demo
Operating System
windows
Hostname
SAL-06
FQDN
SAL-06.corp.net
Release
Microsoft Windows 7 Professional Service Pack 16.1.7601 Build 7601
Architecture
amd64
Client Metadata
Key
Value
2022-09-23T13:02:20Z


New Hunt - Configure Hunt
Description
Quarantine all suspect systems
Expiry
9/30/2022 10:37 AM
Include Condition
Match by label
Include Labels
compromised
Exclude Condition
Run everywhere
Estimated affected clients 10
All known Clients
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch

Create Hunt: Configure artifact parameters

- Artifact

- Windows.Remediation.Quarantine

PolicyName: VelociraptorQuarantine

RuleLookupTable

Action	SrcAddr	SrcMask	SrcPort	DstAddr	DstMask	DstPort	Protocol	Mirrored	Description
Permit	me		0	any		53	udp	yes	DNS
Permit	me		0	any		53	tcp	yes	DNS TCP
Permit	me		68	any		67	udp	yes	DHCP
Block	any			any				yes	All other traffic

MessageBox: This system is compromised and under investigation. Please contact the SOC.

RemovePolicy: ☐ Tickbox to remove policy.

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch

New Hunt - Configure Hunt

Description: Pull triage forensic data from compromised systems

Expiry: 9/30/2022 10:37 AM

Include Condition: Match by label

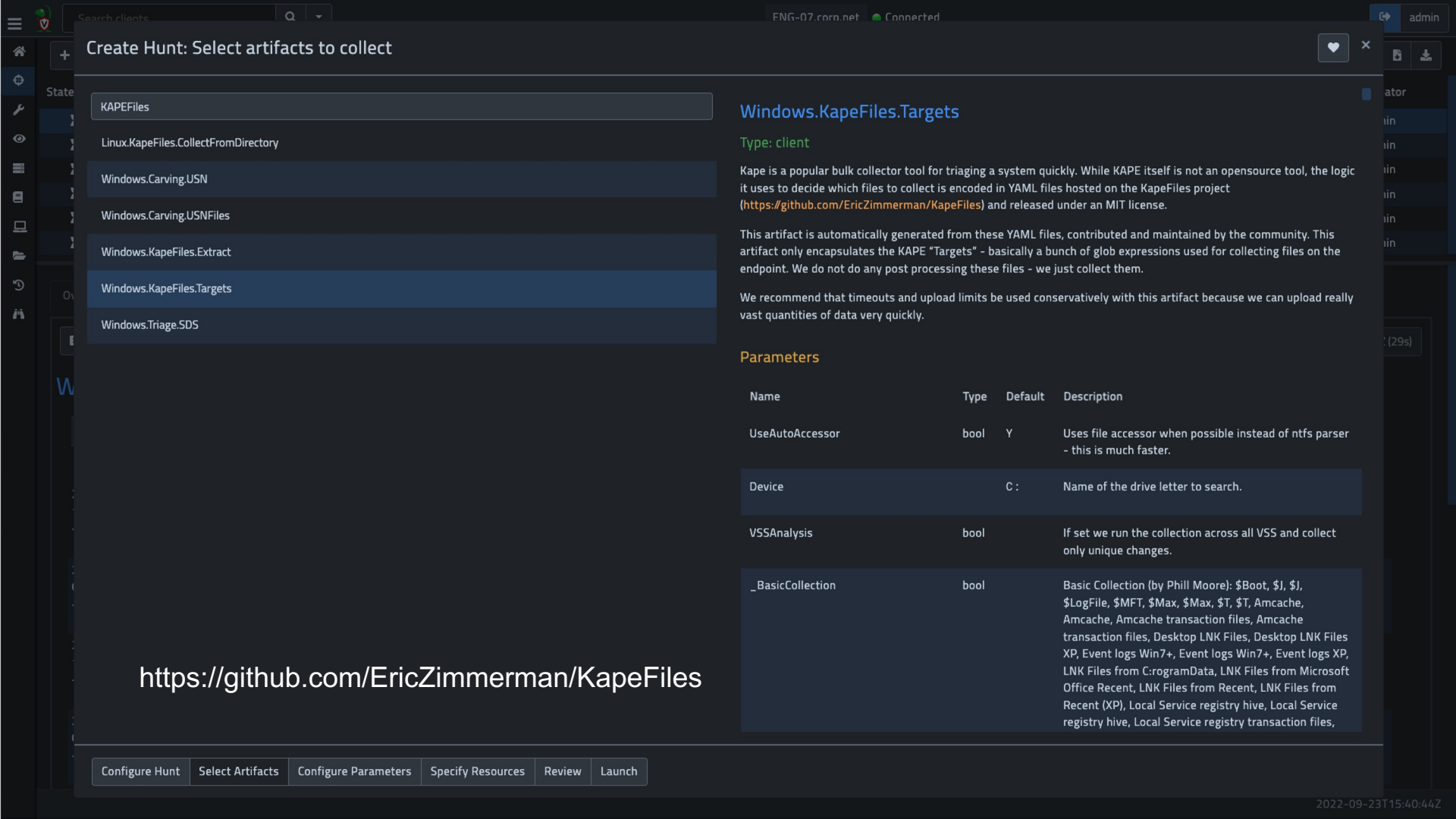
Include Labels: compromised

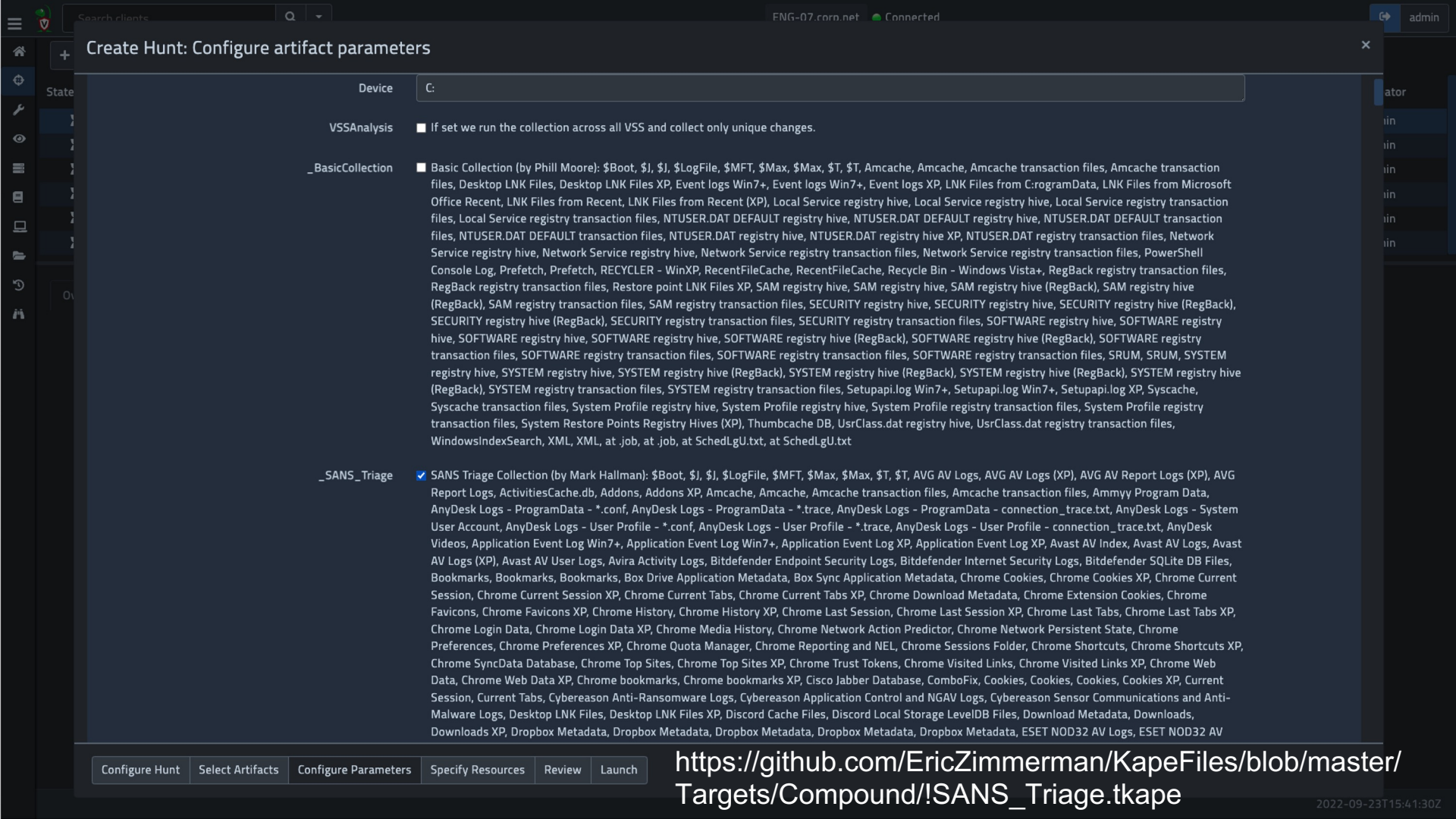
Exclude Condition: Run everywhere

Estimated affected clients 10

All known Clients

Configure Hunt | Select Artifacts | Configure Parameters | Specify Resources | Review | Launch


Create Hunt: Select artifacts to collect
KAPEFiles
Linux.KapeFiles.CollectFromDirectory
Windows.Carving.USN
Windows.Carving.USNFiles
Windows.KapeFiles.Extract
Windows.KapeFiles.Targets
Windows.Triage.SDS
Windows.KapeFiles.Targets
Type: client
Kape is a popular bulk collector tool for triaging a system quickly. While KAPE itself is not an opensource tool, the logic it uses to decide which files to collect is encoded in YAML files hosted on the KapeFiles project (https://github.com/EricZimmerman/KapeFiles) and released under an MIT license.
This artifact is automatically generated from these YAML files, contributed and maintained by the community. This artifact only encapsulates the KAPE "Targets" - basically a bunch of glob expressions used for collecting files on the endpoint. We do not do any post processing these files - we just collect them.
We recommend that timeouts and upload limits be used conservatively with this artifact because we can upload really vast quantities of data very quickly.
Parameters
Name
Type
Default
Description
UseAutoAccessor
bool
Y
Uses file accessor when possible instead of ntfs parser - this is much faster.
Device
C:
Name of the drive letter to search.
VSSAnalysis
bool
If set we run the collection across all VSS and collect only unique changes.
_BasicCollection
bool
Basic Collection (by Phill Moore): $Boot, $J, $J, $LogFile, $MFT, $Max, $Max, $T, $T, Amcache, Amcache, Amcache transaction files, Amcache transaction files, Desktop LNK Files, Desktop LNK Files XP, Event logs Win7+, Event logs Win7+, Event logs XP, LNK Files from C:rogramData, LNK Files from Microsoft Office Recent, LNK Files from Recent, LNK Files from Recent (XP), Local Service registry hive, Local Service registry hive, Local Service registry transaction files,
https://github.com/EricZimmerman/KapeFiles
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch


Create Hunt: Configure artifact parameters
Device
C:
VSSAnalysis
If set we run the collection across all VSS and collect only unique changes.
_BasicCollection
Basic Collection (by Phill Moore): $Boot, $J, $J, $LogFile, $MFT, $Max, $Max, $T, $T, Amcache, Amcache, Amcache transaction files, Amcache transaction files, Desktop LNK Files, Desktop LNK Files XP, Event logs Win7+, Event logs Win7+, Event logs XP, LNK Files from C:rogramData, LNK Files from Microsoft Office Recent, LNK Files from Recent, LNK Files from Recent (XP), Local Service registry hive, Local Service registry hive, Local Service registry transaction files, Local Service registry transaction files, NTUSER.DAT DEFAULT registry hive, NTUSER.DAT DEFAULT registry hive, NTUSER.DAT DEFAULT transaction files, NTUSER.DAT DEFAULT transaction files, NTUSER.DAT registry hive, NTUSER.DAT registry hive XP, NTUSER.DAT registry transaction files, Network Service registry hive, Network Service registry hive, Network Service registry transaction files, Network Service registry transaction files, PowerShell Console Log, Prefetch, Prefetch, RECYCLER - WinXP, RecentFileCache, RecentFileCache, Recycle Bin - Windows Vista+, RegBack registry transaction files, RegBack registry transaction files, Restore point LNK Files XP, SAM registry hive, SAM registry hive, SAM registry hive (RegBack), SAM registry hive (RegBack), SAM registry transaction files, SAM registry transaction files, SECURITY registry hive, SECURITY registry hive, SECURITY registry hive (RegBack), SECURITY registry hive (RegBack), SECURITY registry transaction files, SECURITY registry transaction files, SOFTWARE registry hive, SOFTWARE registry hive, SOFTWARE registry hive, SOFTWARE registry hive, SOFTWARE registry hive (RegBack), SOFTWARE registry hive (RegBack), SOFTWARE registry transaction files, SOFTWARE registry transaction files, SOFTWARE registry transaction files, SOFTWARE registry transaction files, SRUM, SRUM, SYSTEM registry hive, SYSTEM registry hive, SYSTEM registry hive (RegBack), SYSTEM registry hive (RegBack), SYSTEM registry hive (RegBack), SYSTEM registry hive (RegBack), SYSTEM registry transaction files, SYSTEM registry transaction files, Setupapi.log Win7+, Setupapi.log Win7+, Setupapi.log XP, Syscache, Syscache transaction files, System Profile registry hive, System Profile registry hive, System Profile registry transaction files, System Profile registry transaction files, System Restore Points Registry Hives (XP), Thumbcache DB, UsrClass.dat registry hive, UsrClass.dat registry transaction files, WindowsIndexSearch, XML, XML, at .job, at .job, at SchedLgU.txt, at SchedLgU.txt
_SANS_Triage
SANS Triage Collection (by Mark Hallman): $Boot, $J, $J, $LogFile, $MFT, $Max, $Max, $T, $T, AVG AV Logs, AVG AV Logs (XP), AVG AV Report Logs (XP), AVG Report Logs, ActivitiesCache.db, Addons, Addons XP, Amcache, Amcache, Amcache transaction files, Amcache transaction files, Ammyy Program Data, AnyDesk Logs - ProgramData - *.conf, AnyDesk Logs - ProgramData - *.trace, AnyDesk Logs - ProgramData - connection_trace.txt, AnyDesk Logs - System User Account, AnyDesk Logs - User Profile - *.conf, AnyDesk Logs - User Profile - *.trace, AnyDesk Logs - User Profile - connection_trace.txt, AnyDesk Videos, Application Event Log Win7+, Application Event Log Win7+, Application Event Log XP, Application Event Log XP, Avast AV Index, Avast AV Logs, Avast AV Logs (XP), Avast AV User Logs, Avira Activity Logs, Bitdefender Endpoint Security Logs, Bitdefender Internet Security Logs, Bitdefender SQLite DB Files, Bookmarks, Bookmarks, Bookmarks, Box Drive Application Metadata, Box Sync Application Metadata, Chrome Cookies, Chrome Cookies XP, Chrome Current Session, Chrome Current Session XP, Chrome Current Tabs, Chrome Current Tabs XP, Chrome Download Metadata, Chrome Extension Cookies, Chrome Favicons, Chrome Favicons XP, Chrome History, Chrome History XP, Chrome Last Session, Chrome Last Session XP, Chrome Last Tabs, Chrome Last Tabs XP, Chrome Login Data, Chrome Login Data XP, Chrome Media History, Chrome Network Action Predictor, Chrome Network Persistent State, Chrome Preferences, Chrome Preferences XP, Chrome Quota Manager, Chrome Reporting and NEL, Chrome Sessions Folder, Chrome Shortcuts, Chrome Shortcuts XP, Chrome SyncData Database, Chrome Top Sites, Chrome Top Sites XP, Chrome Trust Tokens, Chrome Visited Links, Chrome Visited Links XP, Chrome Web Data, Chrome Web Data XP, Chrome bookmarks, Chrome bookmarks XP, Cisco Jabber Database, ComboFix, Cookies, Cookies, Cookies, Cookies XP, Current Session, Current Tabs, Cybereason Anti-Ransomware Logs, Cybereason Application Control and NGAV Logs, Cybereason Sensor Communications and Anti-Malware Logs, Desktop LNK Files, Desktop LNK Files XP, Discord Cache Files, Discord Local Storage LevelDB Files, Download Metadata, Downloads, Downloads XP, Dropbox Metadata, Dropbox Metadata, Dropbox Metadata, Dropbox Metadata, Dropbox Metadata, ESET NOD32 AV Logs, ESET NOD32 AV
Configure Hunt
Select Artifacts
Configure Parameters
Specify Resources
Review
Launch
https://github.com/EricZimmerman/KapeFiles/blob/master/Targets/Compound/!SANS_Triage.tkape

```
LET upload_to_s3(ClientId, FlowId, Fqdn) = SELECT ClientId,
       upload_s3(bucket=bucket,
                 credentialskey=credentialskey,
                 credentialssecret=credentialssecret,
                 region=region,
                 file=output_file,
                 name=format(format="Host %v %v %v.zip",
                    args=[Fqdn, FlowId, timestamp(epoch=now())])) AS S3
FROM collect(artifacts="UploadFlow", artifact_definitions=UploadFlowDefinition,
             args=dict(`UploadFlow`=dict(
                    ClientId=ClientId, FlowId=FlowId)),
             output=output_file)

LET completions = SELECT *, client_info(client_id=ClientId).os_info.fqdn AS Fqdn
    FROM watch_monitoring(artifact="System.Flow.Completion")
    WHERE Flow.artifacts_with_results =~ ArtifactNameRegex

SELECT * FROM foreach(row=completions, query={
  SELECT * FROM upload_to_s3(ClientId=ClientId, FlowId=FlowId, Fqdn=Fqdn)
})
```

Ship your triage acquisitions off to be plaso'ed and psort'ed into Timesketch.

ArtifactNameRegex: Windows.KapeFiles.Targets

https://github.com/ReconInfoSec/velociraptor-to-timesketch

Thanks for Attending!

Want to go deeper?

- Incident Response with Velociraptor:
  - https://reconis.co/IR_with_VR
- Breaches Be Crazy:
  - https://reconis.co/breaches

@eric_capuano@infosec.exchange

@shortstack@infosec.exchange

www.reconinfosec.com

github.com/reconinfosec

RECON™ INFOSEC