Network Detection Strategies: MDE and GSA better together

Robbe Van den Daele

Thank you Sponsors

Gold

liquidware

2Pint

PINK

RECAST SOFTWARE

Silver

PATCH MY PC

Technical Partners

CONSULTEQ

PROXSYS

Daalmans consulting

Secure At Work

About me

Focus

Microsoft Security; MITRE ATT&CK

From

Belgium

My Blog

HybridBrothers – hybridbrothers.com

Contributions

Blogposts; Public Speaking; GitHub Projects; MC2MC

Hobbies

Community work; Running; Dog training

Contact

LinkedIn; X (@RobbeVdDaele)

Agenda

Network Security Data

The four categories

MDE and it’s logging

How does MDE do network logging

GSA and it’s logging

How does GSA do network logging

MDE VS GSA

What are the differences

Proxy logging and SSL/TLS Inspections

What do we expect

Key takeaways:

- **Understanding Network Security Data**
- **Understand differences between MDE and GSA**
- **Understand how they can work together**

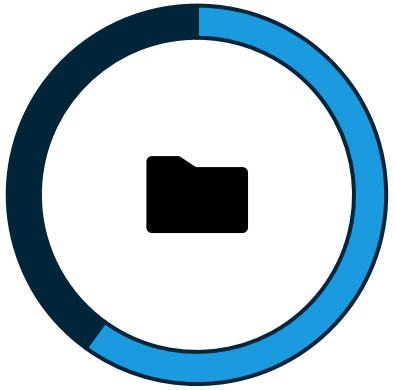
Network Security Data

The four categories

Types of Network Security Data

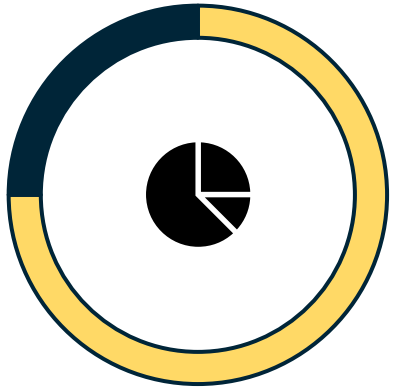
Full content

Also called ‘recorded traffic’, like PCAP files



Extracted Content

Extracts files from the network to feed them into analysis engines



Transaction data

Summarized data traffic

Alert data

‘Judges’ traffic and throws alerts like an NIDS system

Defender for Endpoint and Zeek

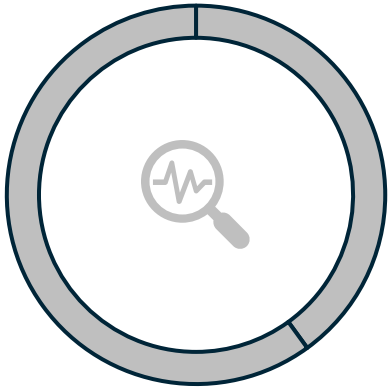
How does MDE do network logging

Network Analysis Engine → Zeek

- Defender for Endpoint → Zeek Engine
- Open-source Network Security Monitoring Tool
- Known for
  - Transaction data
  - Extracted Content data
- Judgement free and policy neutral by default

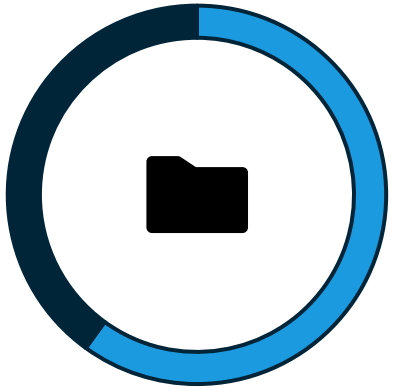
zeek®

Zeek network data



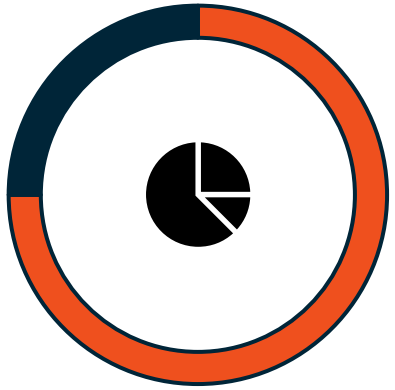
Full content

Also called ‘recorded traffic’, like PCAP files



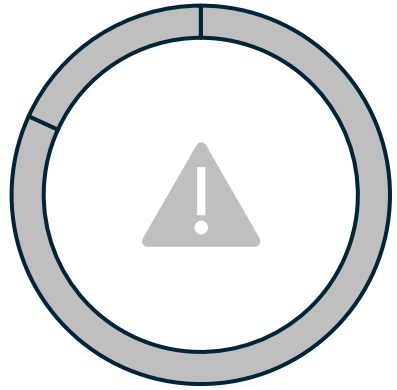
Extracted Content

Extracts files from the network to feed them into analysis engines



Transaction data

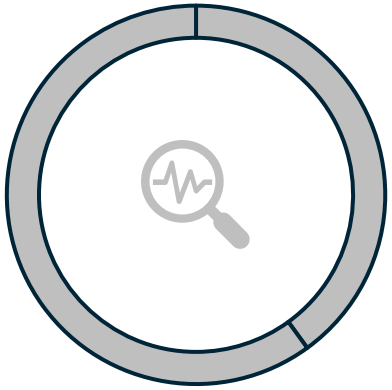
Summarized data traffic



Alert data

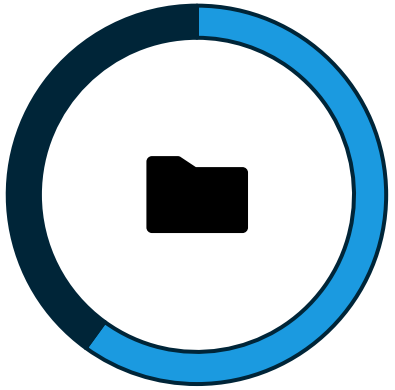
‘Judges’ traffic and throws alerts like an NIDS system

MDE implementation



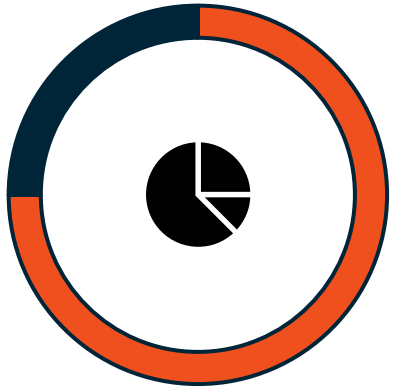
Full content

Also called ‘recorded traffic’, like PCAP files



Extracted Content

Extracts files from the network to feed them into analysis engines



Transaction data

Summarized data traffic

Alert data

‘Judges’ traffic and throws alerts like an NIDS system

Quote

"Zeek has some capability to perform classical byte-centric intrusion detection, but that job is best suited for packages like the open-source **Snort** or **Suricata** engines"

MDE and Zeek assumptions

- MDE Alert implementation not known
  - Zeek Notice mechanism?
  - Zeek data feed into custom engine of MDE?
- Transaction data for Advanced Hunting logging
- Extracted Content for file analysis

MDE – Transaction Data Logging

How does MDE do network logging

Zeek log types

- Multiple default Zeek log types
- Connection streams in ‘conn.log’
- Specific protocols have their own log
- Correlation between conn.log and others

Zeek Logs

- conn.log
- dns.log
- http.log
- files.log
- ftp.log
- ssl.log
- x509.log
- smtp.log
- ssh.log
- pe.log
- dhcp.log
- ntp.log
- SMB Logs (plus DCE-RPC, Kerberos, NTLM)
- irc.log
- ldap.log and ldap_search.log
- postgresql.log
- quic.log
- rdp.log
- traceroute.log
- tunnel.log
- dpd.log
- known_*.log and software.log
- weird.log and notice.log
- capture_loss.log and reporter.log

Zeek connection log

- One of the basic and most important Zeek logs
- Tracks both TCP and UDP connections
- **UID field** for each event to correlate ‘connections’ in other transaction logs
  - Correlate connection to specific protocol logs
- The UID field present in around **20%** of Advanced Hunting logs

```
1   let total = toscalar (
2       DeviceNetworkEvents
3       | summarize count()
4   );
5   DeviceNetworkEvents
6   | extend AdditionalFields = todynamic(AdditionalFields)
7   | extend ConnectionUid = tostring(AdditionalFields.uid)
8   | summarize count() by ConnectionUid
9   | sort by count_ desc
10  | extend Percent = todecimal(count_) / total * 100
11  | top 100 by count_
```

ConnectionUid	count_	Percent
	13723102	80.38246139541346
C3g5L02aLyeTqCR...	8316	0.04871060121569149
CPmPUl35feMRVP...	8273	0.048458730622584864
C1O2lF3JKzfTRwE...	8258	0.04837086878778022
CeVez8PlwuP4b3...	8215	0.04811899819467359
CvlFWu15VAZDKR...	8089	0.04738095878231463
CTkV8q1WL11Mg...	7968	0.046672206648223885
CLbz2a22FwntDA...	7967	0.04666634919257024

Zeek connection log

- UID field only seem present in *ConnectionInspected Action Types (Layer 7 protocol matches)
- Loses its value of correlating events between log types

```
1   DeviceNetworkEvents
2   | extend AdditionalFields = todynamic(AdditionalFields)
3   | extend ConnectionUid = tostring(AdditionalFields.uid)
4   | extend ConnUidPresent = iff(ConnectionUid == "", "False", "True")
5   | summarize count() by ActionType, ConnUidPresent
```

ActionType	ConnUidPresent	count_
ConnectionAcknowledged	False	728361
ConnectionAttempt	False	23325
ConnectionFailed	False	104321
ConnectionFound	False	1335
ConnectionRequest	False	20306
ConnectionSuccess	False	1795861
DnsConnectionInspected	True	492552
FtpConnectionInspected	True	57
HttpConnectionInspected	False	64462
IcmpConnectionInspected	True	15129
InboundConnectionAccepted	False	22672
ListeningConnectionCreated	False	43204
NetworkSignatureInspected	False	529143
NtlmAuthenticationInspected	True	66003
SshConnectionInspected	True	4214
SslConnectionInspected	True	359956

Zeek connection log

- Some correlations can be found when combining with Local IP

```
DeviceNetworkEvents
| extend AdditionalFields = todynamic(AdditionalFields)
| extend ConnectionUid = tostring(AdditionalFields.uid)
| where ConnectionUid == "<uid>" and LocalIP == "<source_ip>"
```

Timestamp	LocalIP	LocalPort	RemoteIP	RemotePort	Protocol	ActionType
Jul 19, 2024 8:26:22 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:22 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:22 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:22 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:22 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected
Jul 19, 2024 8:26:23 AM	10.101.100.93	51869	10.230.1.10	53	Udp	DnsConnectionInspected

MDE and Zeek UID challenges

- Only **20%** contains UID → Pivoting hard to perform
- UID only present in protocol logs
- When used with LocalIP, we do find related traffic

Which Zeek logs is MDE using?

- Comparing Zeek documentation with ActionType in DeviceNetworkEvents

Zeek Logs

- conn.log
- dns.log
- http.log
- files.log
- ftp.log
- ssl.log
- x509.log
- smtp.log
- ssh.log
- pe.log
- dhcp.log
- ntp.log
- SMB Logs (plus DCE-RPC, Kerberos, NTLM)
- irc.log
- rdp.log
- ldap.log and ldap_search.log
- quic.log
- traceroute.log
- tunnel.log
- dpd.log
- known_*.log and software.log
- weird.log and notice.log
- capture_loss.log and reporter.log

DeviceNetworkEvents
Advanced table

See preview data

ActionType values

ConnectionAcknowledged
An acknowledgement that a TCP connection was accepted (syn/ack) was made.

ConnectionAttempt
An attempt to establish a TCP connection (syn) was made.

ConnectionFailed
An attempt to establish a network connection from the device failed.

ConnectionFound
An active network connection was found on the device.

ConnectionRequest
The device initiated a network connection.

ConnectionSuccess
A network connection was successfully established from the device.

DnsConnectionInspected
The deep packet inspection engine in Microsoft Defender for Endpoint inspected a DNS connection.

FtpConnectionInspected
The deep packet inspection engine in Microsoft Defender for Endpoint

Which Zeek logs is MDE using?

- Not all Zeek log files seem to be integrated in MDE
- NetworkSignatureInspected ActionType contains more info with specific protocol logging

```
1  DeviceNetworkEvents
2  | where ActionType == "NetworkSignatureInspected"
3  | extend AdditionalFields = todynamic(AdditionalFields)
4  | extend Signature = tostring(AdditionalFields.SignatureName)
5  | distinct Signature
```

Signature
SMB_Client
Kerberos_TGS_REQ
NTLM-Challenge
HttpServerHeader
CVE-2021-44228

Zeek Logs	DeviceNetworkEvents ActionType
conn.log	Multiple
dns.log	DnsConnectionInspected
http.log	HttpConnectionInspected
files.log	FtpConnectionInspected
ftp.log	FtpConnectionInspected
x509.log	SslConnectionInspected
ssl.log	SslConnectionInspected
smtp.log	SmtpConnectionInspected
ssh.log	SshConnectionInspected
pe.log	NA
dhcp.log	NA
ntp.log	NA
smb.log	NetworkSignatureInspected
ntlm.log	NtlmAuthenticationInspected
irc.log	NA
rdp.log	NA
ldap.log	NA
quic.log	NA
traceroute.log	NA
tunnel.log	NA
dpd.log	NA

MDE Zeek config

- Zeek config can be found at “C:\ProgramData\Microsoft\Windows Defender Advanced Threat Protection\SenseNDR\zeek”

cve	29/02/2024 14:24	File folder
frameworks	29/02/2024 14:24	File folder
policy	29/02/2024 14:24	File folder
protocols	29/02/2024 14:24	File folder
signatures	29/02/2024 14:24	File folder
common_imports.zeek	01/10/2023 13:58	ZEEK File
main.zeek	01/10/2023 13:58	ZEEK File
main_imports.zeek	01/10/2023 13:58	ZEEK File

```
module SMBGhost;

# OriginalCompressedSegmentSize
const compressed_size_field_start:count = 8;
const compressed_size_field_end:count = 12;

# OffsetOrLength
const offset_or_length_field_start:count = 16;
const offset_or_length_field_end:count = 20;

const bytes_to_send:count = 300;

const max:count = 4294967295;

event signature_match(state: signature_state, msg: string, data: string) {
        if(msg != "smb_ghost_exploit")
        {
        return;
        }

        if(|data| < offset_or_length_field_end + 1)
        {
        return;
        }

        local compressedSize: count = bytestring_to_count(data[SMBGhost::compressed_size_field_start:SMBGhost::compressed_size_field_end]);
        local offset_or_length: count = bytestring_to_count(data[SMBGhost::offset_or_length_field_start:SMBGhost::offset_or_length_field_end]);
        if(compressedSize + offset_or_length > SMBGhost::max)
        {
                local payload: string = data;
                local paylod_cutoff: bool = F;
                if(|payload| > SMBGhost::bytes_to_send)
                {
                        payload = payload[:bytes_to_send];
                        paylod_cutoff = T;
                }

                local info: Signatures::Info = [$ts=network_time(),
                                                $uid=state$conn$uid,
                                                $id=state$conn$id,
                                                $proto=get_port_transport_proto(state$conn$id$resp_p),
```

MDE – SSL/TLS Inspection

Analyzing certificate usage

HTTP VS HTTPS

- HTTPS = HTTP over TLS(or SSL)
- TLS is a TCP/IP Layer 5 protocol (Application Layer)
- Uses **encryption** to provide confidentiality, integrity, and authenticity
- In December 2022, **58.4%** of the Internet's most popular websites use HTTPS (HTTPS – Wikipedia)

- ➔ TLS Decryption used in network security services to inspect encrypted traffic

Does Zeek support TLS Decryption?

- Yes! But …
  - Only TLS 1.2
  - Only cipher ‘TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384’
  - Zeek is not planning to extend support (according to documentation)

- ➔ How often is this cipher used?

TLS Cipher usage

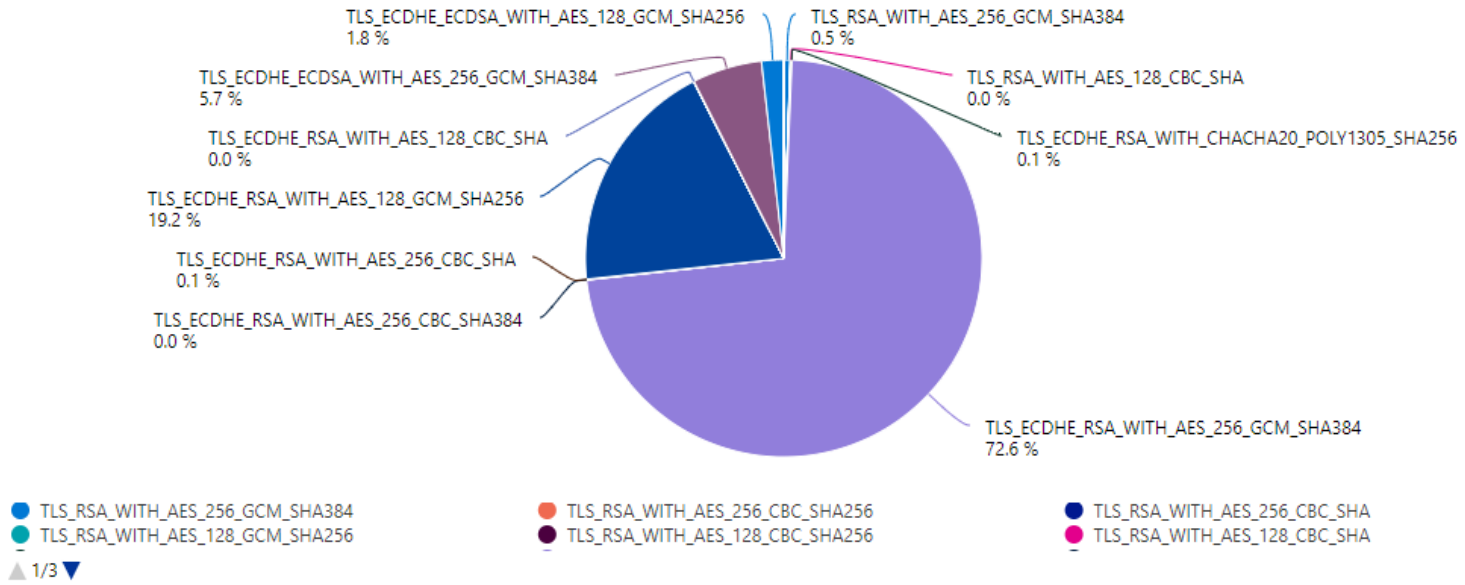
- Analyzing SSL connections using ‘SslConnectionInspected’ ActionType in DeviceNetworkEvents
- ➔ Most of the traffic is supported by Zeek for decryption

```
1   // SSL inspections
2   DeviceNetworkEvents
3   | where ActionType == "SslConnectionInspected"
4   | extend AdditionalFields = todynamic(AdditionalFields)
5   | extend Direction = tostring(AdditionalFields.direction)
6       , Version = tostring(AdditionalFields.version)
7       , FQDN = tostring(AdditionalFields.server_name)
8       , Cipher = tostring(AdditionalFields.cipher)
9   // Get traffic that Zeek is theoretically able to decrypt https://docs.zeel
10  | summarize count() by Version, Cipher
11  | sort by count_ desc
```

Version	Cipher	count_
TLSv12	TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384	260835
TLSv12	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	50789
TLSv12	TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	31841
TLSv12	TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	13667
TLSv12	TLS_RSA_WITH_AES_128_CBC_SHA	369
TLSv12	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA	347
TLSv12	TLS_RSA_WITH_AES_128_GCM_SHA256	285
TLSv12	TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256	266
TLSv12	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA	251
TLSv12	TLS_RSA_WITH_AES_256_GCM_SHA384	218
DTLSv12	TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	110
TLSv12	TLS_RSA_WITH_AES_256_CBC_SHA	85
TLSv12	TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384	35
TLSv12	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256	31
TLSv12	TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	13
TLSv10	TLS_RSA_WITH_AES_256_CBC_SHA	3

TLS Cipher usage

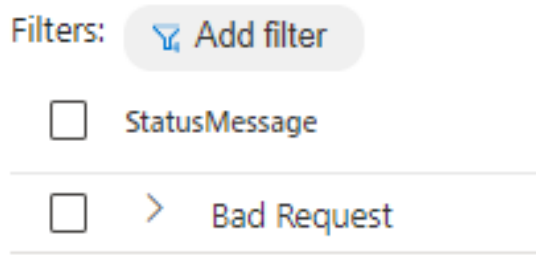
- Crafting some insights
- Tested on 10 organizations
- Between **57,1%** and **81,4%** of SSL Certificates used the ‘TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384’ cipher
- **Average of 72,28%**



Does MDE support TLS Decryption?

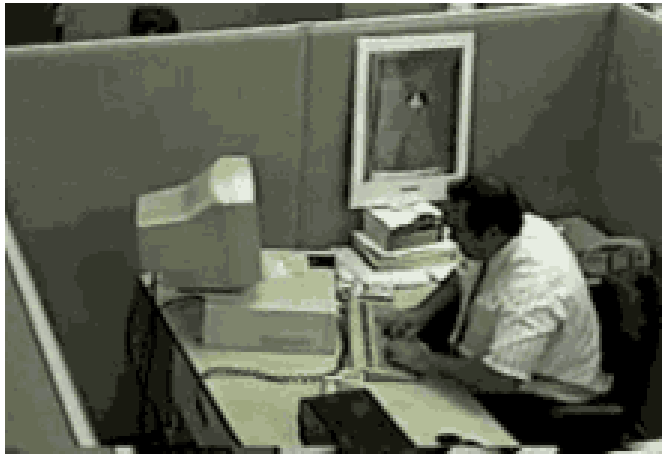
- No ☹
  - Not event the cipher supported by Zeek

```
15  DeviceNetworkEvents
16  | where ActionType contains "HTTPConnection"
17  | where RemotePort == "443"
18  | extend StatusMessage = tostring(todynamic(AdditionalFields).status_msg)
19  | distinct StatusMessage
```



- HTTPS only logged as connections, without extra HTTP header info

- ➔ Probably because TLS decryption is resource intensive



Does MDE support TLS Decryption?

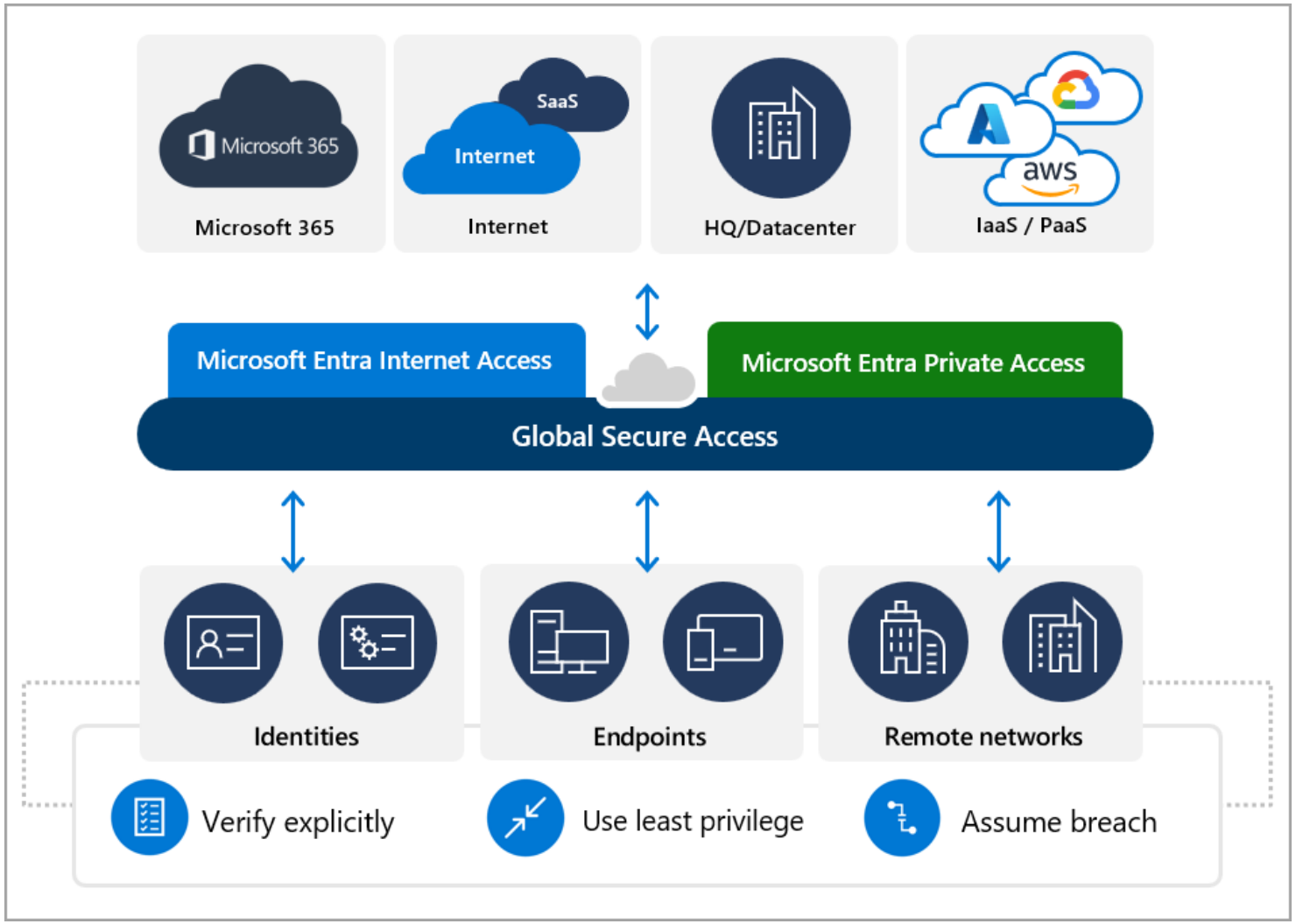
- What is the solution to have TLS Decryption?

- ➔ Proxy (SSE) Solution – Zscaler, Palo Alto Prisma, Cato Networks
- ➔ Microsoft Entra Global Secure Access

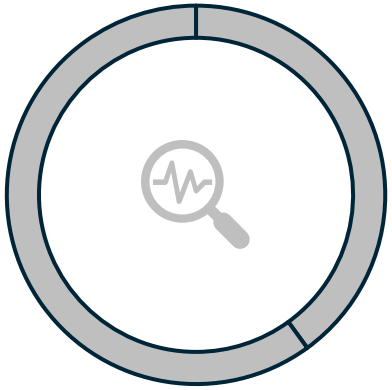
Global Secure Access

How does GSA do network logging

Focus on Internet Access

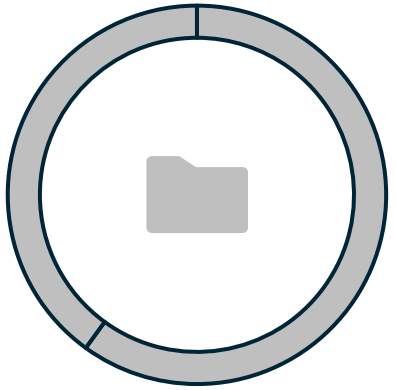


GSA current logging capabilities



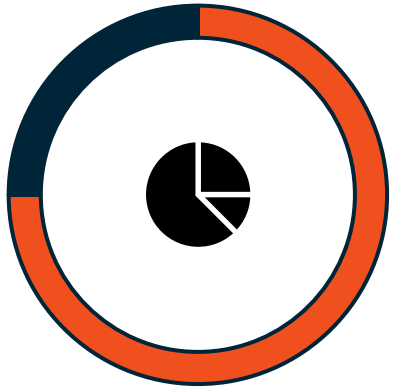
Full content

Also called ‘recorded traffic’, like PCAP files



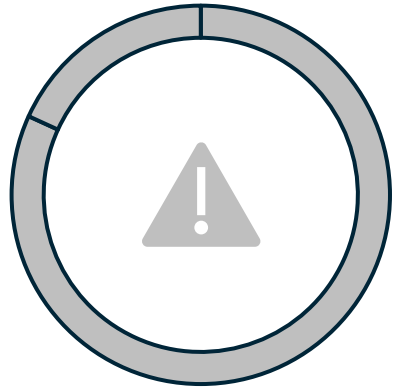
Extracted Content

Extracts files from the network to feed them into analysis engines



Transaction data

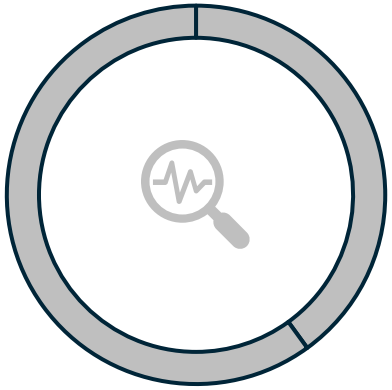
Summarized data traffic



Alert data

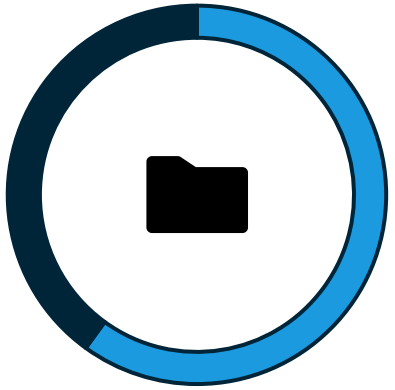
‘Judges’ traffic and throws alerts like an NIDS system

GSA theoretical logging capabilities



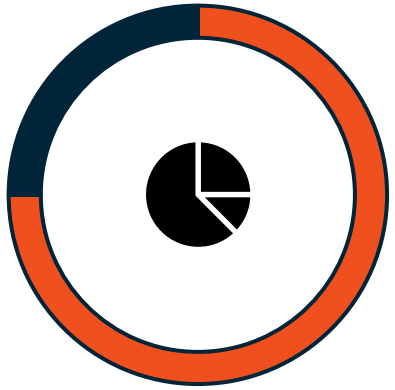
Full content

Also called ‘recorded traffic’, like PCAP files



Extracted Content

Extracts files from the network to feed them into analysis engines



Transaction data

Summarized data traffic

Alert data

‘Judges’ traffic and throws alerts like a NIDS system

GSA Traffic logs

- Three categories – Connections; Transactions; Sessions

- Connection – A TCP/UDP connection between two endpoints identified by 5-tuple
  - Source IP, Source Port, Destination IP, Destination Port, Protocol
  - Unique identification ➔ ConnectionID
- Transaction – Activity or operation within the Connection
  - Identified by 5-tuple + HTTP request/response
  - Unique identification → TransactionID
  - Multiple transactions can be opened per connection
  - Applicable for Internet Access and M365 traffic
- Session – Logical aggregation of multiple connections and transactions associated with a specific website (first URL a user accesses)

GSA Traffic logs

- Multiple logs with same connectionId

```
1  NetworkAccessTraffic
2  | summarize count() by ConnectionId, SourceIp, SourcePort, DestinationIp, DestinationPort, NetworkProtocol
```

ConnectionId	SourceIp	SourcePort	DestinationIp	DestinationPort	NetworkProtocol	count_
5EGeII/yl06Z5H7H.0	213.118.[redacted]	49796	151.101.38.172	80	Ipv4	23
mbyo8eLSZ0OQFtGy.0	213.118.[redacted]	49887		443	Ipv4	7
GAexdcKpbESuggMj.0	213.118.[redacted]	49887		443	Ipv4	7
y6RVrigw9EaZJeIn.0	213.118.[redacted]	49887		443	Ipv4	7
O6vMHo0sdUqPn1Xu.0	213.118.[redacted]	49887		443	Ipv4	6

- Multiple logs with unique transactionId

```
4  NetworkAccessTraffic
5  | where ConnectionId == "5EGeII/yl06Z5H7H.0"
6  | project TransactionId, SentBytes, ReceivedBytes, HttpMethod, ResponseCode
```

TransactionId	SentBytes	ReceivedBytes	HttpMethod	ResponseCode
130574ba-3c45-4638-85af-aef22064f0a3	0	0	Head	200
65c543b1-807f-4512-bd24-b092a1301fd5	0	1120	Get	206
50f8f3e2-df5f-4ea4-879e-8689ca9ec31b	0	306	Get	206
341257e2-a0b8-4148-aef5-020a02be5d94	0	1252	Get	206
f2618098-88d2-41af-af24-ca69bceb3c6d	0	4614	Get	206
a2904a9e-4f33-4f72-b1d8-07a7aa65f56a	0	1521	Get	206

HTTP logging with proxy solutions

Which logging data do we expect?

Interesting HTTP Data

- Source @Cybe3rMonk

- Typical detections:
  - Find Beaconing
  - C2 activity
  - Data exfiltration
  - Malicious payload delivery
  - Malicious files
  - Malicious binaries
  - Malicious URLs
  - Malicious traffic
  - Old protocols

Web Proxy Hunting and Detection Cheat Sheet
Version 1.0
Mehmet Ergene @Cyb3rMonk

Attribute	Technique	What to look for
Duration	Calculate the sum per SourceIP-DestinationIP pair over 12/24 hours	Higher values may indicate beaconing
HTTP Status Code	Calculate the total count of the HTTP Status Codes per SourceIP or per SourceIP-DestinationIP over a specific time period.	Higher values of an uncommon HTTP Status Code may indicate C2 activity.
	List URLs having only HTTP Errors	C2 servers may rotate their dns records, malware tries every domain and causes http errors.
Bytes In	Calculate the count of BytesIn per Source-Destination pair over 12/24 hours	Higher values may indicate beaconing. C2 servers reply with the same data, making Bytes In value the same
	Calculate the ratio of count(BytesIn) per Source-Destination pair	Higher values of ratio may indicate beaconing
Bytes Out	Calculate the sum of BytesOut per Source-Destination pair over 12/24 hours	Higher values may indicate data exfiltration
	Calculate the ratio of count(BytesOut) per Source-Destination pair over 12/24 hours	Higher values of ratio may indicate beaconing
HTTP Method	Calculate the ratio of POST or PUT over GET per Source-Destination over 4/8/12/24 hours	Higher values may indicate beaconing or exfiltration
URL Hostname	Compare with top 1M domains and calculate hit count	Hit count <5 and Hostname is not in top 1M may indicate malicious payload delivery
	Calculate hit count per Hostname	Less hit count may indicate malicious payload delivery
URL Path	Calculate count per Source-Destination-URLPath pair	Higher values may indicate beaconing
URL Query	Calculate count per Source-Destination-URLQuery	Higher values may indicate beaconing
	Calculate length of URLQuery	Higher values may indicate encoded data, a sign of exfiltration or beaconing
	Look for base64 encoded strings in URLQuery	Encoded strings may indicate beaconing or exfiltration
Content Type	List Content Type per Source-Destination pair	Uncommon Content types may indicate malicious file
User Agent	Calculate count within the environment(long tail analysis)	Lower values may indicate a malicious binary
URL Category	Query for Uncategorized, Dynamic DNS, and other suspicious categories. Calculate dcount of SourceAddress by URLHostname	Small dcount values may indicate abnormal/suspicious/malicious activity. If an uncategorized URL is visited by many users, it is less likely that the URL is malicious.
HTTP Version	Check HTTP versions	1.0 is older, might be suspicious
Protocol	Compare ports with protocols	Common Protocol-Uncommon Port or Common Port-Uncommon Protocol may indicate malicious traffic
File Name	Entropy analysis on filenames.	May indicate malicious payload delivery

What does MDE have?

- HTTP logs can be found with ActionType “HttpConnectionInspected” in DeviceNetworkEvents

direction	Out
host	www.microsoft.com
method	GET
request_body_len	0
response_body_len	1456
status_code	200
status_msg	OK
tags	[]
trans_depth	1
uri	/pkiops/certs/Microsoft Azure RSA TLS Issuing CA 03 - xsign.crt
user_agent	Microsoft-CryptoAPI/10.0
version	1.1

What does GSA have?

- DestinationFQDN
- DestinationUrl
- DestinationWebCategories
- HTTP Method
- OriginHeader
- ReferrerHeader
- ReceivedBytes
- SentBytes
- ThreatType
- XForwardedFor

MS Learn table reference

MDE VS GSA HTTP Logging

MDE Event capping makes this unreliable

- Together we have almost every important HTTP log
- This only applies for HTTP traffic

Expected HTTP Logging data	MDE - HttpConnectionInspected Columns	GSA - NetworkTrafficLogs Columns
Duration	Not available	Not available; Although we might be able to calculate it via ConnectionStatus column (not verified)
HTTP Status	AdditionalFields.status_code	ResponseCode
Bytes In	AdditionalFields.response_body_len	ReceivedBytes
Bytes Out	AdditionalFields.response_body_len	SentBytes
HTTP Method	AdditionalFields.method	HTTP Method
URL Hostname	AdditionalFields.host	DestinationFqdn
URL Path	AdditionalFields.uri	DestinationUrl
MIME(Content) type	Not available (only in FtpConnectionInspected)	Not available
User Agent	AdditionalFields.user_agent	Not available
URL Category	Not available	DestinationWebCategories
HTTP Version	AdditionalFields.version	Not available
Protocol	ActionType	TrafficType
File Name	Can be found via other MDE tables	Not available
Threat Type	Not available	ThreatType
Referrer Header	Not available	ReferrerHeader
XForwardedFor	Not available	XForwardedFor

Remember TLS Decryption?

- MDE Does not support TLS Decryption
- Does GSA Support it? → **Not Yet**
- When it does, table scheme should be more complete imho → We cannot rely on MDE for missing fields
- → This is why NetworkTrafficLogs have empty columns for HTTPS traffic

Run | Time range : Last 24 hours | Save | Share

```
1  NetworkAccessTraffic
2  | where DestinationPort == 80
3  | project HttpMethod, ResponseCode, XForwardedFor
```

Results | Chart | Add bookmark

HttpMethod	ResponseCode	XForwardedFor
Get	304	213.118.119.195
Get	304	213.118.119.195
Get	200	213.118.119.195

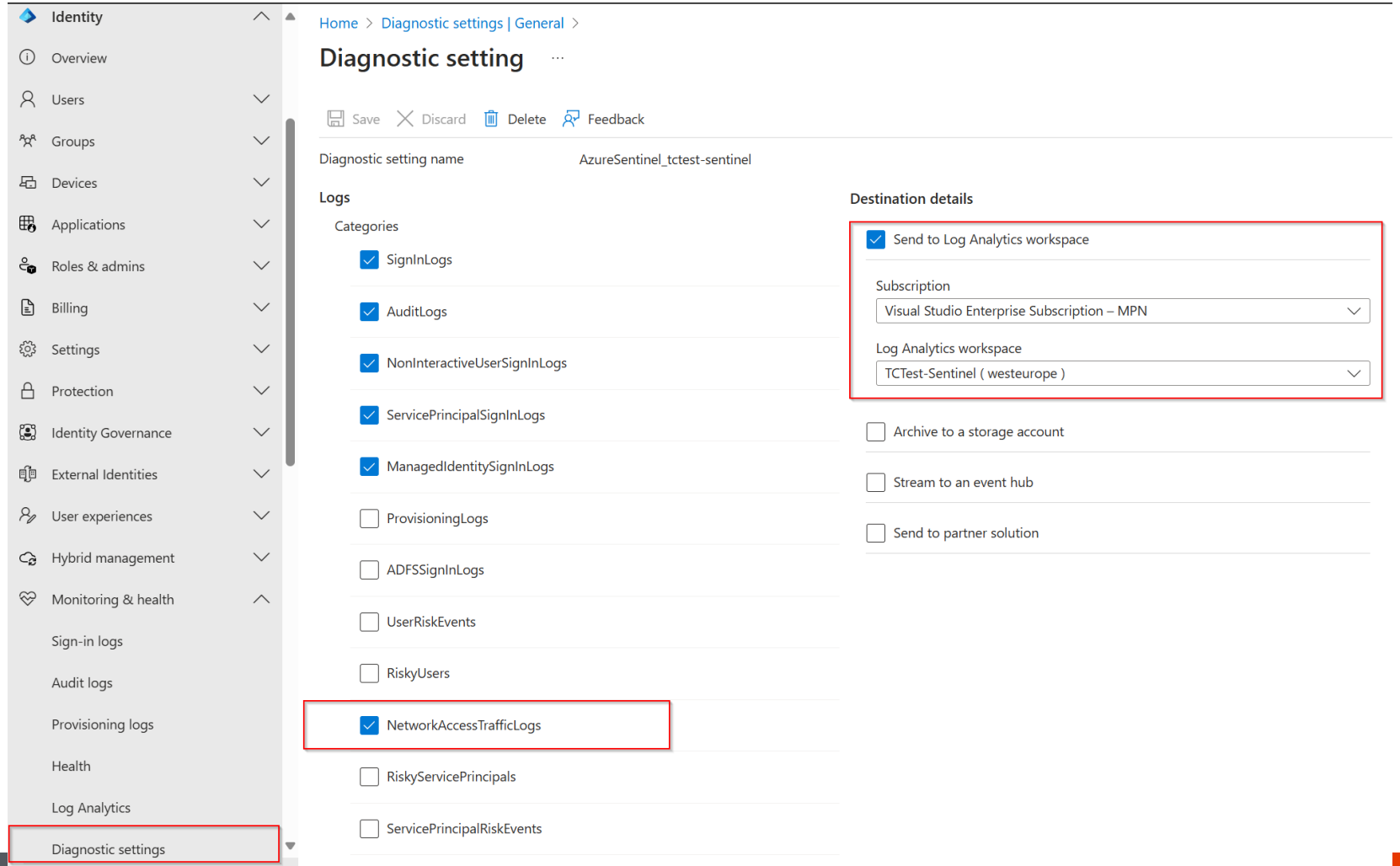
```
1  NetworkAccessTraffic
2  | where DestinationPort == 443
3  | project HttpMethod, ResponseCode, XForwardedFor
```

Results | Chart | Add bookmark

HttpMethod	ResponseCode	XForwardedFor

How to correlate GSA with MDE logs?

Step 1 – Ingest NetworkAccessTrafficLogs in Microsoft Sentinel (Identity > Monitoring & health > Diagnostic Settings)



How to correlate GSA with MDE logs?

Step 2 – Ingest DeviceNetworkEvent in Microsoft Sentinel to correlate in Sentinel (more expensive) or enable USOP platform to correlate in Defender XDR

Microsoft Defender XDR

Connected
Status

Microsoft
Provider

Description
Microsoft Defender XDR is a unified, natively in
breach enterprise defense suite that protects e
and applications and helps you detect, prevent
automatically respond to sophisticated threats.

Microsoft Defender XDR suite includes:

- Microsoft Defender for Endpoint
- Microsoft Defender for Identity
- Microsoft Defender for Office 365
- Microsoft Defender for Cloud Apps
- Microsoft Defender Alerts
- Microsoft Defender Vulnerability Managen
- Microsoft Purview Data Loss Prevention
- Microsoft Entra ID Protection

When turning on Incidents synchronization **Mi**
Cloud Incidents are automatically synced to Se
To match all alerts that may appear in incidents
Microsoft Defender for Cloud connector to b
their entire collection of subscriptions in the te

Last data received
--

Actions & submissions
Partner catalog
Threat intelligence
Assets
Microsoft Sentinel
Identities
Endpoints
Email & collaboration
Cloud apps
Cases
SOC optimization
Reports
Learning hub
Trials
More resources
System
Audit
Permissions
Settings

Settings

Microsoft Sentinel
Workspaces
Settings for 'TCTest-Sentinel'
Pricing
Entity behavior analytics
Anomalies
Workspace manager configuration
Playbook permissions
Auditing and health monitoring
Remove Microsoft Sentinel
Log Analytics settings

Workspaces

Currently only one workspace can be connected

Select the Microsoft Sentinel workspace you'd like to connect to Microsoft Defender XDR. Learn more about connecting a workspace

Connect workspace
Search
Filter

Filters: Subscription: Azure subscription 1 +4

Name	Status	Resource Group	Location	Subscription
TCTest-Sentinel	Connected	secops	westeurope	Visual Studio Enterprise Subscriptic

ed to Sentinel.

How to correlate GSA with MDE logs?

- No uniform ‘transactionId’ between MDE logs and GSA logs

- ➔ Try to correlate the ‘5-tuple connections’ from MDE with GSA
  - SourcePort
  - SourceIP
  - DestinationPort
  - DestinationIP
  - Protocol

How to correlate GSA with MDE logs?

- Challenges
  - Source IP not the same in MDE and GSA ➔ Use DeviceId instead

```
69   NetworkAccessTraffic
70   | where * contains "httpforever.com"
71   | distinct SourceIp
72
```

Getting started | Results | Query history

Export | Show empty columns

Filters: Add filter

SourceIp
213.118.1[redacted]

```
69   DeviceNetworkEvents
70   | where * contains "httpforever.com"
71   | distinct LocalIP
72
```

Getting started | Results | Query history

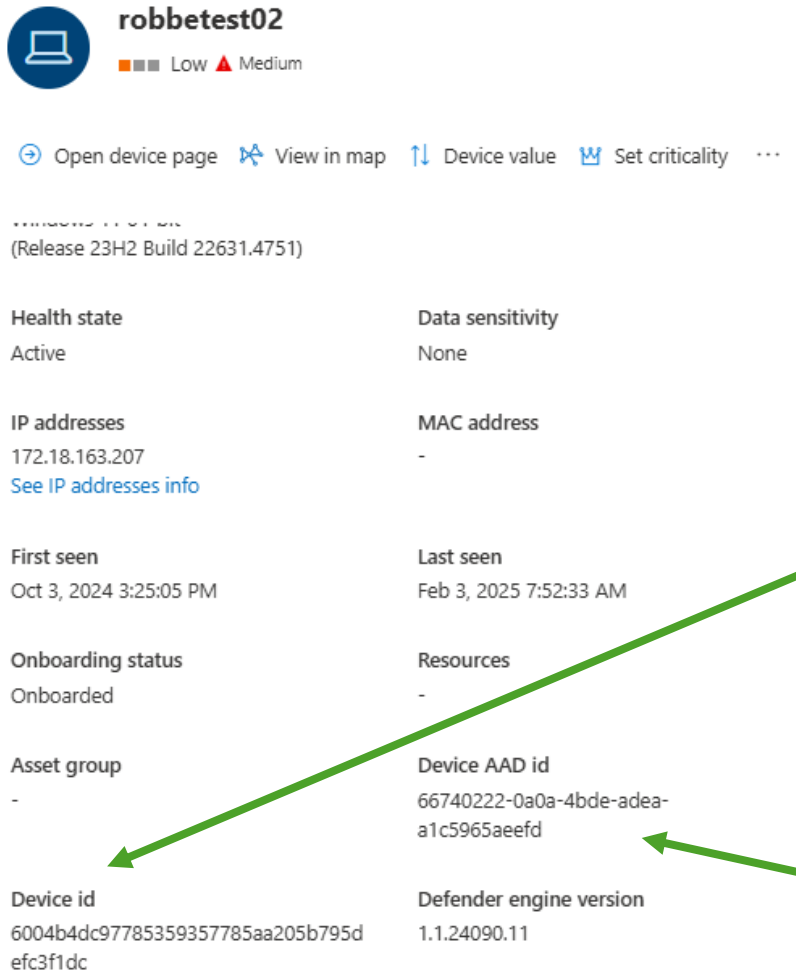
Export | Show empty columns

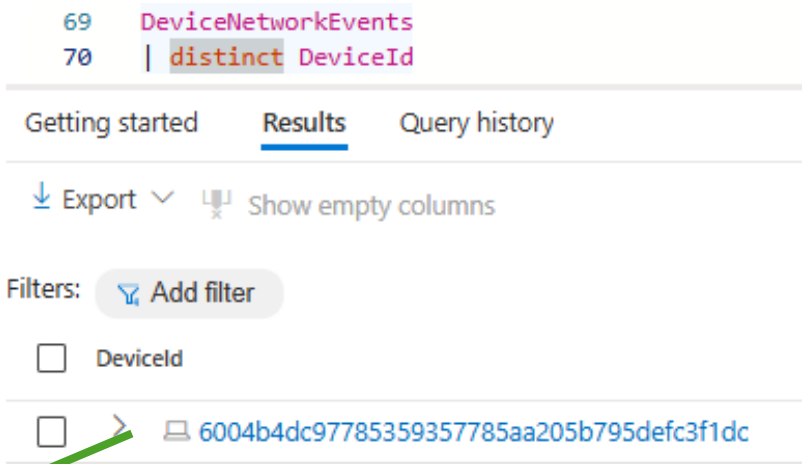
Filters: Add filter

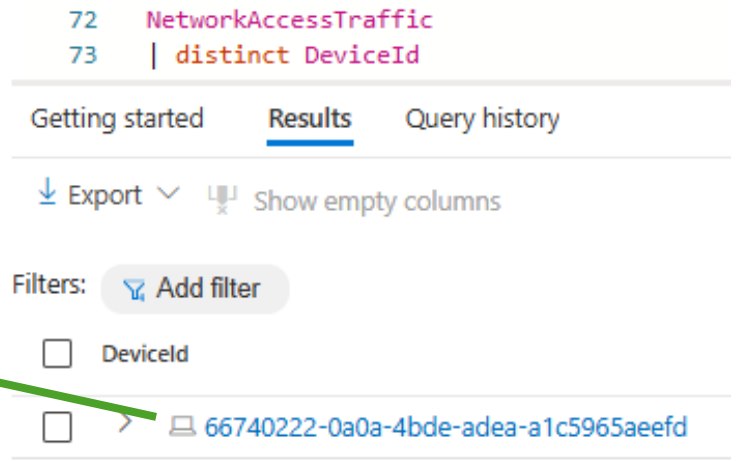
LocalIP
172.18.163.207

How to correlate GSA with MDE logs?

- Challenges
  - Source IP not the same i
  - Device ID not the same







How to correlate GSA with MDE logs?

- Challenges
  - Source IP not the same in MDE and G
  - Device ID not the same

```
C:\Users\RobbeVandenDaele>nslookup httpforever.com
Server:  TC-00WUQ23053FB.mshome.net
Address:  172.18.160.1

Non-authoritative answer:
Name:    httpforever.com
Address:  6.6.0.236
```

Global Secure Access Client - Advanced diagnostics

Overview | Health check | Forwarding profile | Hostname acquisition | Traffic | Advanced log collection

Network traffic

Collect and analyze this device's network traffic. Learn more about Traffic page

Start collecting | Export CSV | Clear table | Add filter | Columns

Protocol	Destination FQDN	Destination IP	Destination port	Correlation vector ID	Process name	Bytes sent
Udp		172.18.160.1	53		svchost.exe	0
Tcp	httpforever.com	6.6.0.236	80	MAxo/3f92Uyvb65V.0	msedge.exe	770
Tcp	httpforever.com	6.6.0.236	80	2EHWjMqh6kmZP7bI.0	msedge.exe	92
Tcp	httpforever.com	6.6.0.236	443	D58R07wAFUywESWZ.0	msedge.exe	2119

```
68  NetworkAccessTraffic
69  | where * contains "httpforever.com"
70  | distinct DestinationFqdn, DestinationIp
```

Getting started | Results | Query history

Export | Show empty columns

Filters: Add filter

DestinationFqdn	DestinationIp
httpforever.com	146.190.62.39

```
68  DeviceNetworkEvents
69  | where * contains "httpforever.com"
70  | distinct RemoteUrl, RemoteIP
71
```

Getting started | Results | Query history

Export | Show empty columns

Filters: Add filter

RemoteUrl	RemoteIP
httpforever.com	6.6.0.236

How to correlate GSA with MDE logs?

- Challenges
  - Source IP not the same in MDE and GSA ➔ Use DeviceId instead
  - Device ID not the same
  - Remote IP and Destination IP not the same due to GSA tunnel
  - LocalPort mapping not reliable due to MDE capping

```
68  DeviceNetworkEvents
69  | where * contains "httpforever.com"
70  | distinct LocalPort
71
```

Getting started | Results | Query history

Export | Show empty columns

Filters: Add filter

LocalPort
55366
56111
56110
65216

```
69  NetworkAccessTraffic
70  | where * contains "httpforever.com"
71  | distinct SourcePort
72
```

Getting started | Results | Query history

Export | Show empty columns

Filters: Add filter

SourcePort
51582
49864

How to correlate GSA with MDE logs?

- What can we do?
  - Enrich DeviceId of GSA logs with DeviceInfo table
  - Correlate on DeviceId
  - Correlate on DestinationFQDN
  - Correlate on DestinationPort
  - Correlate on Protocol
  - Correlate on InitiatingProcess

How to correlate GSA with MDE logs?

```
let gsa_events = NetworkAccessTraffic
    // Join DeviceInfo to get MDE DeviceID
    | join kind=inner (
        DeviceInfo
        | distinct DeviceId, AadDeviceId
    ) on $left.DeviceId == $right.AadDeviceId
    // Remove Entra Device ID from GSA logs
    | project-away DeviceId
    // Rename MDE Device ID to DeviceId column
    | project-rename DeviceId = DeviceId1;
// Get all MDE network events
DeviceNetworkEvents
// Get HTTP details if HTTP connection is logged
| extend HttpStatus = toint(todynamic(AdditionalFields).status_code),
    BytesIn = toint(todynamic(AdditionalFields).response_body_len),
    BytesOut = toint(todynamic(AdditionalFields).request_body_len),
    HttpMethod = tostring(todynamic(AdditionalFields).method),
    UrlHostname = tostring(todynamic(AdditionalFields).host),
    UrlPath = tostring(todynamic(AdditionalFields).uri),
    UserAgent = tostring(todynamic(AdditionalFields).user_agent),
    HttpVersion = tostring(todynamic(AdditionalFields).version)
// Join GSA logs
| join kind=inner gsa_events on
    DeviceId,
    $left.RemoteUrl == $right.DestinationFqdn,
    $left.RemotePort == $right.DestinationPort,
    $left.Protocol == $right.TransportProtocol,
    $left.InitiatingProcessFileName == $right.InitiatingProcessName
| project-rename TimeGeneratedGsa = TimeGenerated1, TimestampMde = Timestamp
| project-away Type, TenantId, TimeGenerated, TenantId1, Type1, DeviceId1, AadDeviceId
```

Extra benefits

- Detailed Process logging of MDE with detailed network logging of GSA!

➔Great for detection engineering

➔Allows to reduce BP rate

UserId
03dd93c4-dee8-44d6-ad99-baf2644d63f3

UserPrincipalName
robbe.van.den.daele@thecollectivetest.eu

TransportProtocol
Tcp

NetworkProtocol
Ipv4

Action
Allow

SentBytes
0

ReceivedBytes
0

ReferrerHeader
https://www.bing.com/

XForwardedFor
213.118.119.195

InitiatingProcessName
msedge.exe

DestinationUrl
http://httpforever.com/

Inspect record

InitiatingProcessSHA1
c646d9b8426a0d7b4d6a2adf3cbdcd145254bd00

InitiatingProcessSHA256
89f47856d2ce6a173e97c3f6b6a7724cc9279fbc5b7dbd763a059e3230...

InitiatingProcessMD5
7f0fec67361e0512bca1d5c4f8bff9a1

InitiatingProcessFileName
msedge.exe

InitiatingProcessFileSize
3923496

InitiatingProcessVersionInfoCompanyName
Microsoft Corporation

InitiatingProcessVersionInfoProductName
Microsoft Edge

InitiatingProcessVersionInfoProductVersion
132.0.2957.127

InitiatingProcessVersionInfoInternalFileName
msedge_exe

InitiatingProcessVersionInfoOriginalFileName
msedge.exe

InitiatingProcessVersionInfoFileDescription
Microsoft Edge

InitiatingProcessId
8932

InitiatingProcessCommandLine
"msedge.exe" --type=utility --utility-sub-type=network.mojom.NetworkServ...

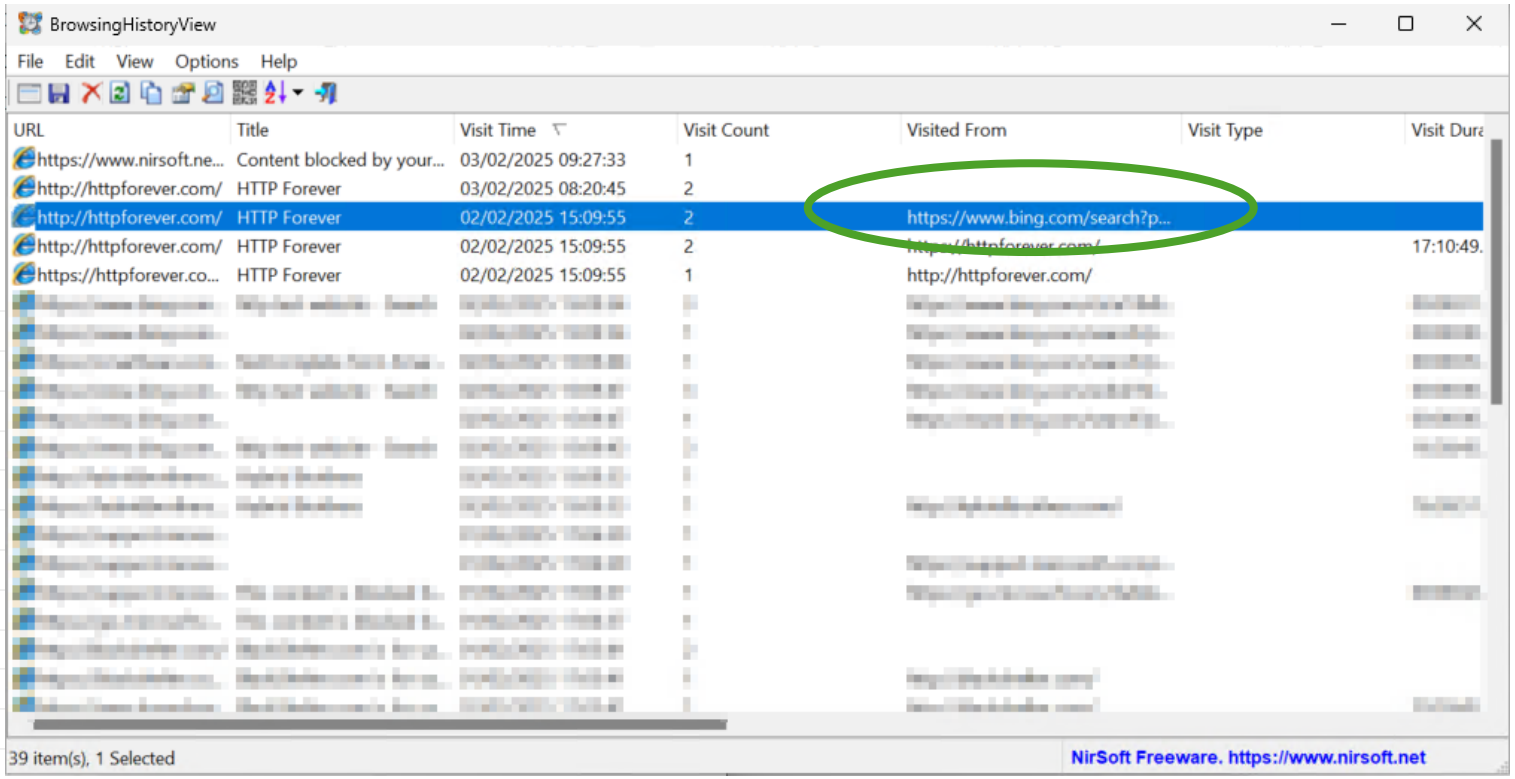
InitiatingProcessCreationTime
Feb 1, 2025 4:46:05 PM

InitiatingProcessFolderPath

Extra benefits

- Threat Hunting
  - MDE process logs do not give details about redirection (only source process)
  - ➔ Browser history dump needed in threat hunting

DeviceName	robbetest02
ActionType	ConnectionSuccess
RemoteIP	6.6.0.236
RemotePort	80
RemoteUrl	httpforever.com
LocalIP	172.18.163.207
LocalPort	56111
Protocol	Tcp
LocalIPType	Private
RemoteIPType	Public
InitiatingProcessSHA1	c646d9b8426a0d7b4d6a2adf3cbdcd145254bd00
InitiatingProcessSHA256	89f47856d2ce6a173e97c3f6b6a7724cc9279fbc5b7dbd763a059e323068a98e
InitiatingProcessMD5	7f0fec67361e0512bca1d5c4f8bff9a1
InitiatingProcessFileName	msedge.exe
InitiatingProcessFileSize	3923496
InitiatingProcessVersionI...	Microsoft Corporation
InitiatingProcessVersionI...	Microsoft Edge
InitiatingProcessVersionI...	132.0.2957.127
InitiatingProcessVersionI...	msedge_exe
InitiatingProcessVersionI...	msedge.exe
InitiatingProcessVersionI...	Microsoft Edge
InitiatingProcessId	8932
InitiatingProcessComma...	"msedge.exe" --type=utility --utility-sub-type=network.mojom.NetworkService --lang=en-GB --service-sandbox-type=none --string-annotations --subproc-heap-profiling



Extra benefits

- Threat Hunting
  - MDE process logs do not give details about redirection (only source process)
  - ➔ Browser history dump needed in threat hunting
  - With GSA, we can see the source pages in RefererHeader!

ReferrerHeader	https://www.bing.com/
XForwardedFor	213.118.119.195
InitiatingProcessName	msedge.exe
DestinationUrl	http://httpforever.com/
HttpMethod	Get
ResponseCode	304
Type	NetworkAccessTraffic

MDE and GSA differences

How do they differ from each other?

Telemetry capping

- MDE caps network events
  - A lot of events have a cap of 1 event per 24 hour
  - Capping based on specific field mappings
  - Reference FalconForce

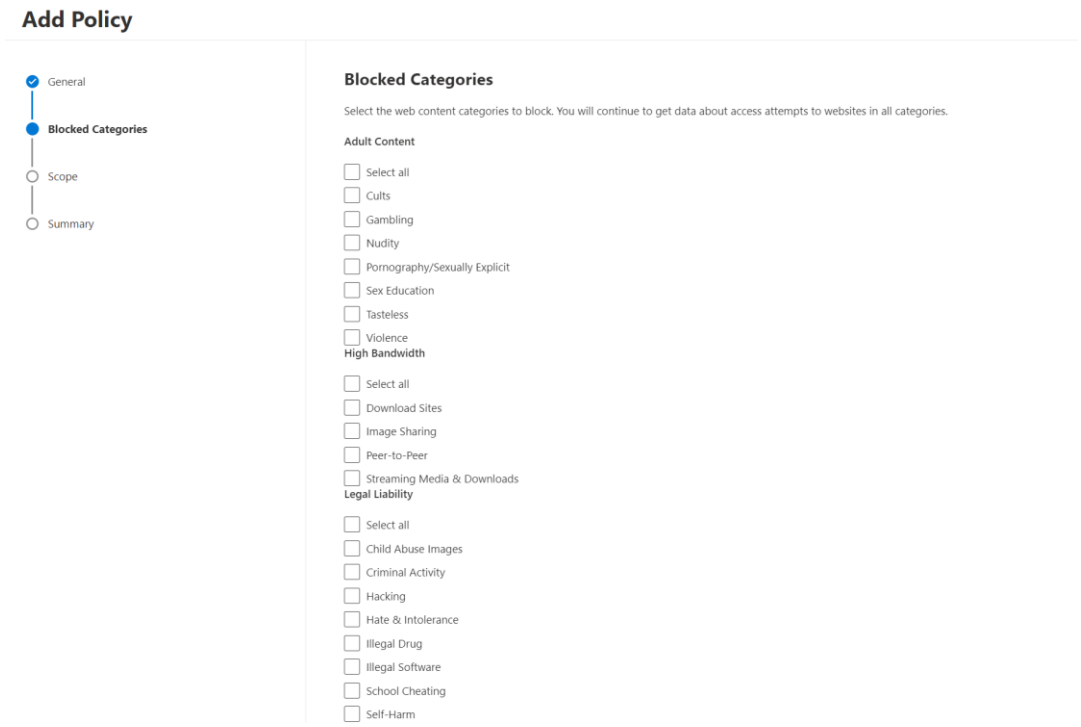
- MDE “HttpConnectionInspected” ActionType is capped more than “ConnectionSuccess”

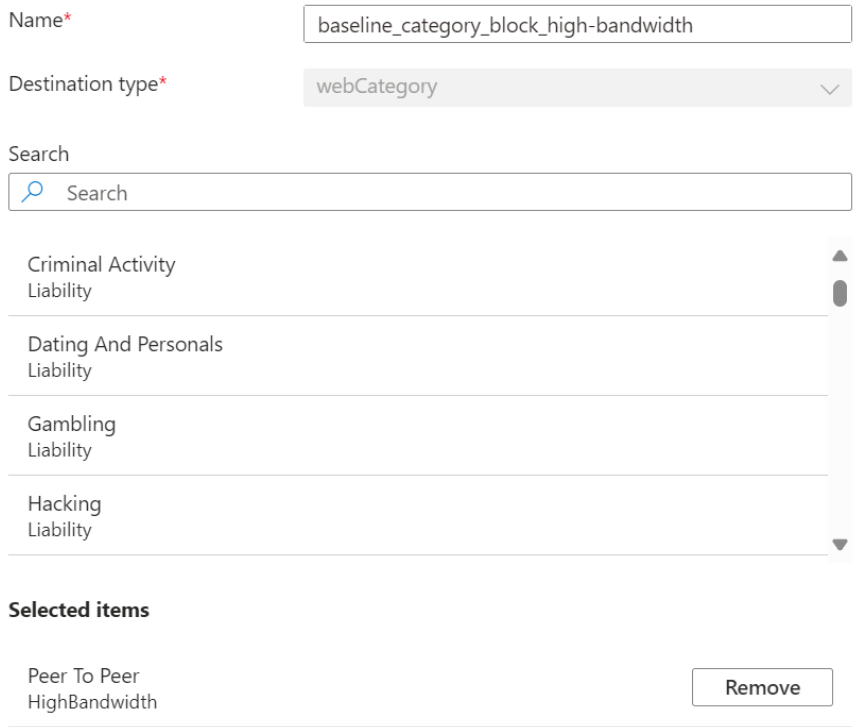
- → This is not the case with GSA


```
{
  "name": "Connect Complete",
  "eventId": 1033,
  "aggregation": {
    "type": "NoAggregation"
  },
  "id": "{B245978C-395F-43F1-9227-ADD41E37AF85}",
  "version": "1",
  "filters": {},
  "capping": {
    "globalCapping": {
      "capping": 4000
    },
    "localCapping": [
      {
        "id": "ConnectCompleteFirstSeen",
        "expirationPeriodInHours": 24,
        "fields": [
          {
            "fieldName": "ExtractedLocalAddress"
          },
          {
            "fieldName": "ExtractedRemoteAddress"
          },
          {
            "fieldName": "ExtractedRemoteAddressPort"
          },
          {
            "fieldName": "InitiatingProcess:PROCESS_NAME"
          },
          {
            "fieldName": "InitiatingProcess:PROCESS_IMAGE_ORIGINAL_NAME"
          },
          {
            "fieldName": "InitiatingProcess:PROCESS_CI_SIGNING_LEVEL"
          },
          {
            "fieldName": "InitiatingProcess:PROCESS_FILE_NATIVE_PATH"
          },
          {
            "fieldName": "InitiatingProcess:PROCESS_FILE_SHA1"
          }
        ],
        "capping": 1
      }
    ]
  },
```

Web Category Filtering

- Feature supported in both MDE and GSA





Web Category Filtering

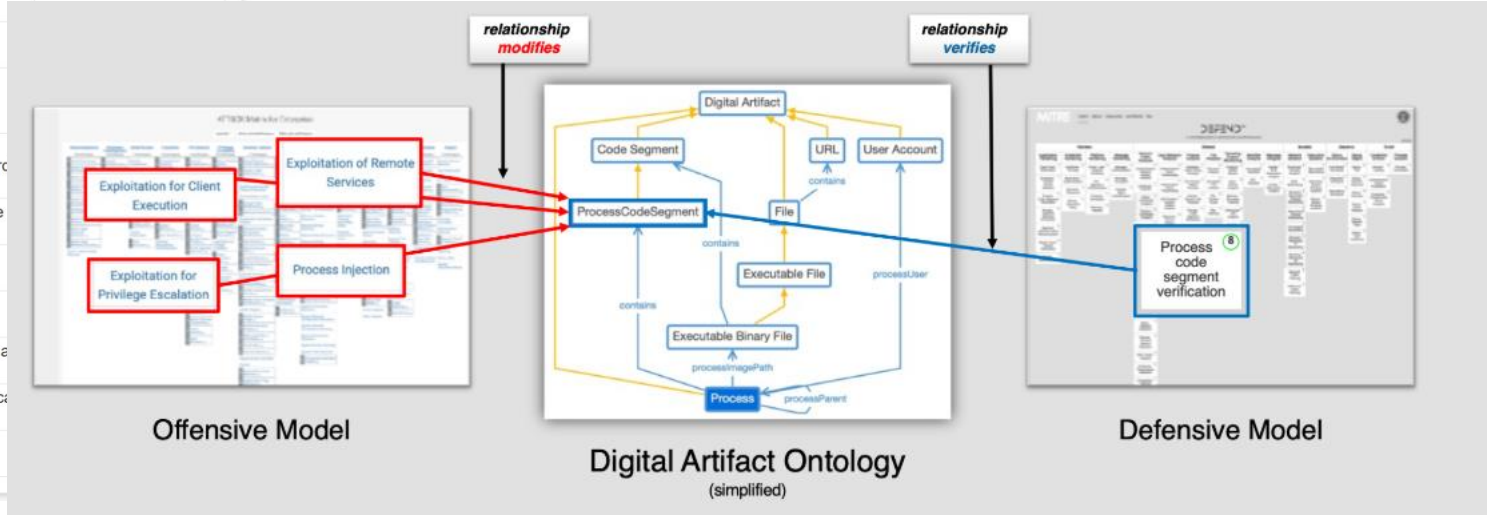
- Similar engine
  - MDE supports smaller list of categories
  - MDE focusses on categories that could introduce liability for the organization operating the site
  - → Many sites are not categorized with MDE Engine

- MDE Web Content Filtering breaks with TLS inspection
  - When using third-party SSE or Firewall

- GSA is more powerful for this

MITRE Mapping

Name	ID	Definition	Synonyms
Network Traffic Analysis	D3-NTA	Analyzing intercepted or summarized computer network traffic to detect unauthorized activity.	
- Relay Pattern Analysis	D3-RPA	The detection of an internal host relaying traffic between the internal network and the external network.	Relay Network Detection
- Remote Terminal Session Detection	D3-RTSD	Detection of an unauthorized remote live terminal console session by examining network traffic to a network host.	
- RPC Traffic Analysis	D3-RTA	Monitoring the activity of remote procedure calls in communication traffic to establish standard protocol operations and potential attacker activities.	RPC Protocol Analysis
- Network Traffic Signature Analysis	D3-NTSA	Analyzing network traffic and compares it to known signatures	
- File Carving	D3-FC	Identifying and extracting files from network application protocols through the use of network stream reassembly software.	
- Inbound Session Volume Analysis	D3-ISVA	Analyzing inbound network session or connection attempt volume.	
- IPC Traffic Analysis	D3-IPCTA	Analyzing standard inter process communication (IPC) protocols to detect deviations from normal protocol activity.	IPC Analysis
- Network Traffic Community Deviation	D3-NTCD	Establishing baseline communities of network hosts and identifying statistically divergent inter-community communication.	
- Per Host Download-Upload Ratio Analysis	D3-PHDURA	Detecting anomalies that indicate malicious activity by comparing the amount of data downloaded versus data uploaded by a host.	
- Administrative Network Activity Analysis	D3-ANAA	Detection of unauthorized use of administrative network protocols by analyzing network activity against a baseline.	
- Byte Sequence Emulation	D3-BSE	Analyzing sequences of bytes and determining if they likely represent malicious shellcode.	
- Certificate Analysis	D3-CA	Analyzing Public Key Infrastructure certificates to detect if they have been misconfigured or spoofed using both network traffic, certificate fields and third party logs.	
- Client-server Payload Profiling	D3-CSPP	Comparing client-server request and response payloads to a baseline profile to identify outliers.	
- Connection Attempt Analysis	D3-CAA	Analyzing failed connections in a network to detect unauthorized activity.	
- DNS Traffic Analysis	D3-DNSTA	Analysis of domain name metadata, including name and DNS records, to determine whether the domain is likely to resolve to an undesirable host.	
- Passive Certificate Analysis	D3-PCA	Collecting host certificates from network traffic or other passive sources like a certificate transparency log and analyzing them for unauthorized activity.	
- Protocol Metadata Anomaly Detection	D3-PMAD	Collecting network communication protocol metadata and identifying statistical outliers.	
- Active Certificate Analysis	D3-ACA	Actively collecting PKI certificates by connecting to the server and downloading its server certificates for analysis.	



- MITRE ATT&CK Hands-Off assessment
- Mapped MDE and GSA via D3FEND framework on Network Traffic Analysis
- Assigned confidence score (0-5) for each subclass

MITRE Mapping

- Why? Trying to give a visual representation of how both solutions cover offensive techniques

- Two mappings

- Mapping One – Total score calculation based on Average of all scores
- Mapping Two – Total score calculation based on Count of defensive mappings

MITRE Mapping

- Each subclass has its own layer
- Both mapped for MDE and GSA
- Scores assigned to relevant techniques

d3-anaa-gsa.json	28/01/2025 16:42	JSON Source File	4 KB
d3-anaa-mde.json	28/01/2025 16:43	JSON Source File	4 KB
d3-caa-gsa.json	28/01/2025 16:48	JSON Source File	6 KB
d3-caa-mde.json	28/01/2025 16:49	JSON Source File	6 KB
d3-ca-gsa.json	28/01/2025 16:44	JSON Source File	3 KB
d3-ca-mde.json	28/01/2025 16:45	JSON Source File	3 KB
d3-cspp-gsa.json	28/01/2025 16:46	JSON Source File	23 KB
d3-cspp-mde.json	28/01/2025 16:47	JSON Source File	23 KB
d3-dnsta-gsa.json	28/01/2025 16:50	JSON Source File	2 KB
d3-dnsta-mde.json	28/01/2025 16:50	JSON Source File	2 KB
d3-fc-gsa.json	28/01/2025 16:35	JSON Source File	2 KB
d3-fc-mde.json	28/01/2025 16:35	JSON Source File	2 KB
d3-ipcta-gsa.json	28/01/2025 16:38	JSON Source File	2 KB
d3-ipcta-mde.json	28/01/2025 16:39	JSON Source File	2 KB
d3-isva-gsa.json	28/01/2025 16:36	JSON Source File	3 KB
d3-isva-mde.json	28/01/2025 16:37	JSON Source File	3 KB
d3-ntcd-gsa.json	28/01/2025 16:40	JSON Source File	23 KB
d3-ntcd-mde.json	28/01/2025 16:40	JSON Source File	23 KB
d3-ntsa-gsa.json	28/01/2025 16:33	JSON Source File	23 KB
d3-ntsa-mde.json	28/01/2025 16:34	JSON Source File	23 KB
d3-phdura-gsa.json	28/01/2025 16:41	JSON Source File	23 KB
d3-phdura-mde.json	28/01/2025 16:42	JSON Source File	23 KB
d3-rpa-gsa.json	28/01/2025 16:28	JSON Source File	9 KB
d3-rpa-mde.json	28/01/2025 16:28	JSON Source File	9 KB
d3-rta-gsa.json	28/01/2025 16:31	JSON Source File	2 KB
d3-rta-mde.json	28/01/2025 16:31	JSON Source File	2 KB
d3-rtsd-gsa.json	28/01/2025 16:29	JSON Source File	23 KB
d3-rtsd-mde.json	28/01/2025 16:30	JSON Source File	23 KB

MITRE Mapping One - MDE

Reconnaissance
10 techniques

- Active Scanning (0/2)
- Gather Victim Host Information (0/4)
- Gather Victim Identity Information (0/3)
- Gather Victim Network Information (0/6)
- Gather Victim Org Information (0/4)
- Phishing for Information (0/3)
- Search Closed Sources (0/2)
- Search Open Technical Databases (0/5)
- Search Open Websites/Domains (0/2)
- Search Victim-Owned Websites

Resource Development
7 techniques

- Acquire Infrastructure (0/6)
- Compromise Accounts (0/2)
- Compromise Infrastructure (0/6)
- Develop Capabilities (0/4)
- Establish Accounts (0/2)
- Obtain Capabilities (0/6)
- Stage Capabilities (0/5)

Initial Access
9 techniques

- Drive-by Compromise
- Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing (2/3)
- Replication Through Removable Media
- Supply Chain Compromise (0/3)
- Trusted Relationship
- Valid Accounts (0/4)

Execution
12 techniques

- Command and Scripting Interpreter (0/8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (0/2)
- Native API
- Scheduled Task/Job (0/7)
- Shared Modules
- Software Deployment Tools
- System Services (0/2)
- User Execution (1/3)
- Windows Management Instrumentation

Persistence
19 techniques

- Account Manipulation (1/4)
- BITS Jobs
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (0/3)
- Create or Modify System Process (0/4)
- Event Triggered Execution (2/15)
- External Remote Services
- Hijack Execution Flow (0/11)
- Implant Internal Image
- Modify Authentication Process (0/4)
- Office Application Startup (0/6)
- Pre-OS Boot (1/5)
- Scheduled Task/Job (0/7)
- Server Software Component (0/3)
- Traffic Signaling (1/1)
- Valid Accounts (0/4)

Privilege Escalation
13 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Create or Modify System Process (0/4)
- Domain Policy Modification (0/2)
- Escape to Host
- Event Triggered Execution (2/15)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (0/11)
- Process Injection (0/11)
- Scheduled Task/Job (0/7)
- Valid Accounts (0/4)

Defense Evasion
39 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- BITS Jobs
- Build Image on Host
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (0/2)
- Execution Guardrails (0/1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (0/2)
- Hide Artifacts (0/7)
- Hijack Execution Flow (0/11)
- Impair Defenses (0/7)
- Indicator Removal on Host (0/6)
- Indirect Command Execution
- Masquerading (0/6)
- Modify Authentication Process (0/4)
- Modify Cloud Compute Infrastructure (0/4)
- Modify Registry
- Modify System Image (0/2)
- Network Boundary Bridging (0/1)
- Obfuscated Files or Information (0/5)
- Pre-OS Boot (1/5)
- Process Injection (0/11)
- Rogue Domain Controller
- Rootkit
- Signed Binary Proxy Execution (1/11)
- Signed Script Proxy Execution (0/1)
- Subvert Trust Controls (0/6)
- Template Injection
- Traffic Signaling (1/1)
- Trusted Developer Utilities Proxy Execution (0/1)
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material (2/4)
- Valid Accounts (0/4)

Credential Access
15 techniques

- Brute Force (2/4)
- Credentials from Password Stores (0/5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials (0/2)
- Input Capture (0/4)
- Man-in-the-Middle (1/2)
- Modify Authentication Process (0/4)
- Network Sniffing
- OS Credential Dumping (1/8)
- Steal Application Access Token
- Steal or Forge Kerberos Tickets (1/4)
- Steal Web Session Cookie
- Two-Factor Authentication Interception
- Unsecured Credentials (0/7)

Discovery
27 techniques

- Account Discovery (0/4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Container and Resource Discovery
- Domain Trust Discovery
- File and Directory Discovery
- Network Service Scanning
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (0/3)
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery (0/1)
- System Information Discovery
- System Location Discovery
- System Network Configuration Discovery (0/1)
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion (0/3)

Lateral Movement
9 techniques

- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (0/2)
- Remote Services (2/6)
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material (2/4)

Collection
17 techniques

- Archive Collected Data (0/3)
- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Cloud Storage Object
- Data from Configuration Repository (0/2)
- Data from Information Repositories (0/2)
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged (0/2)
- Email Collection (0/3)
- Input Capture (0/4)
- Man in the Browser
- Man-in-the-Middle (1/2)
- Screen Capture
- Video Capture

Command and Control
16 techniques

- Application Layer Protocol (4/4)
- Communication Through Removable Media
- Data Encoding (0/2)
- Data Obfuscation (0/3)
- Dynamic Resolution (0/3)
- Encrypted Channel (2/2)
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- Non-Standard Port
- Protocol Tunneling
- Proxy (4/4)
- Remote Access Software
- Traffic Signaling (1/1)
- Web Service (0/3)

Exfiltration
9 techniques

- Automated Exfiltration (0/1)
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol (3/3)
- Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium (0/1)
- Exfiltration Over Physical Medium (0/1)
- Exfiltration Over Web Service (2/2)
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
13 techniques

- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation (1/3)
- Defacement (0/2)
- Disk Wipe (0/2)
- Endpoint Denial of Service (1/4)
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service (2/2)
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

MITRE Mapping One - GSA

Reconnaissance
10 techniques

- Active Scanning (0/2)
- Gather Victim Host Information (0/4)
- Gather Victim Identity Information (0/3)
- Gather Victim Network Information (0/6)
- Gather Victim Org Information (0/4)
- Phishing for Information (0/3)
- Search Closed Sources (0/2)
- Search Open Technical Databases (0/5)
- Search Open Websites/Domains (0/2)
- Search Victim-Owned Websites

Resource Development
7 techniques

- Acquire Infrastructure (0/6)
- Compromise Accounts (0/2)
- Compromise Infrastructure (0/6)
- Develop Capabilities (0/4)
- Establish Accounts (0/2)
- Obtain Capabilities (0/6)
- Stage Capabilities (0/5)

Initial Access
9 techniques

- Drive-by Compromise
- Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing (2/3)
- Replication Through Removable Media
- Supply Chain Compromise (0/3)
- Trusted Relationship
- Valid Accounts (0/4)

Execution
12 techniques

- Command and Scripting Interpreter (0/8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (0/2)
- Native API
- Scheduled Task/Job (0/7)
- Shared Modules
- Software Deployment Tools
- System Services (0/2)
- User Execution (1/3)
- Windows Management Instrumentation

Persistence
19 techniques

- Account Manipulation (1/4)
- BITS Jobs
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (0/3)
- Create or Modify System Process (0/4)
- Event Triggered Execution (2/15)
- External Remote Services
- Hijack Execution Flow (0/11)
- Implant Internal Image
- Modify Authentication Process (0/4)
- Office Application Startup (0/6)
- Pre-OS Boot (1/5)
- Scheduled Task/Job (0/7)
- Server Software Component (0/3)
- Traffic Signaling (1/1)
- Valid Accounts (0/4)

Privilege Escalation
13 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Create or Modify System Process (0/4)
- Domain Policy Modification (0/2)
- Escape to Host
- Event Triggered Execution (2/15)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (0/11)
- Process Injection (0/11)
- Scheduled Task/Job (0/7)
- Valid Accounts (0/4)

Defense Evasion
39 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- BITS Jobs
- Build Image on Host
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (0/2)
- Execution Guardrails (0/1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (0/2)
- Hide Artifacts (0/7)
- Hijack Execution Flow (0/11)
- Impair Defenses (0/7)
- Indicator Removal on Host (0/6)
- Indirect Command Execution
- Masquerading (0/6)
- Modify Authentication Process (0/4)
- Modify Cloud Compute Infrastructure (0/4)
- Modify Registry
- Modify System Image (0/2)
- Network Boundary Bridging (0/1)
- Obfuscated Files or Information (0/5)
- Pre-OS Boot (1/5)
- Process Injection (0/11)
- Rogue Domain Controller
- Rootkit
- Signed Binary Proxy Execution (1/11)
- Signed Script Proxy Execution (0/1)
- Subvert Trust Controls (0/6)
- Template Injection
- Traffic Signaling (1/1)
- Trusted Developer Utilities Proxy Execution (0/1)
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material (2/4)
- Valid Accounts (0/4)

Credential Access
15 techniques

- Brute Force (2/4)
- Credentials from Password Stores (0/5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials (0/2)
- Input Capture (0/4)
- Man-in-the-Middle (1/2)
- Modify Authentication Process (0/4)
- Network Sniffing
- OS Credential Dumping (1/8)
- Steal Application Access Token
- Steal or Forge Kerberos Tickets (1/4)
- Steal Web Session Cookie
- Two-Factor Authentication Interception
- Unsecured Credentials (0/7)

Discovery
27 techniques

- Account Discovery (0/4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Container and Resource Discovery
- Domain Trust Discovery
- File and Directory Discovery
- Network Service Scanning
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (0/3)
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery (0/1)
- System Information Discovery
- System Location Discovery
- System Network Configuration Discovery (0/1)
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion (0/3)

Lateral Movement
9 techniques

- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (0/2)
- Remote Services (2/6)
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material (2/4)

Collection
17 techniques

- Archive Collected Data (0/3)
- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Cloud Storage Object
- Data from Configuration Repository (0/2)
- Data from Information Repositories (0/2)
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged (0/2)
- Email Collection (0/3)
- Input Capture (0/4)
- Man in the Browser
- Man-in-the-Middle (1/2)
- Screen Capture
- Video Capture

Command and Control
16 techniques

- Application Layer Protocol (4/4)
- Communication Through Removable Media
- Data Encoding (0/2)
- Data Obfuscation (0/3)
- Dynamic Resolution (0/3)
- Encrypted Channel (2/2)
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- Non-Standard Port
- Protocol Tunneling
- Proxy (4/4)
- Remote Access Software
- Traffic Signaling (1/1)
- Web Service (0/3)

Exfiltration
9 techniques

- Automated Exfiltration (0/1)
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol (3/3)
- Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium (0/1)
- Exfiltration Over Physical Medium (0/1)
- Exfiltration Over Web Service (2/2)
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
13 techniques

- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation (1/3)
- Defacement (0/2)
- Disk Wipe (0/2)
- Endpoint Denial of Service (1/4)
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service (2/2)
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

MITRE Mapping Two - MDE

Reconnaissance
10 techniques

- Active Scanning (0/2)
- Gather Victim Host Information (0/4)
- Gather Victim Identity Information (0/3)
- Gather Victim Network Information (0/6)
- Gather Victim Org Information (0/4)
- Phishing for Information (0/3)
- Search Closed Sources (0/2)
- Search Open Technical Databases (0/5)
- Search Open Websites/Domains (0/2)
- Search Victim-Owned Websites

Resource Development
7 techniques

- Acquire Infrastructure (0/6)
- Compromise Accounts (0/2)
- Compromise Infrastructure (0/6)
- Develop Capabilities (0/4)
- Establish Accounts (0/2)
- Obtain Capabilities (0/6)
- Stage Capabilities (0/5)

Initial Access
9 techniques

- Drive-by Compromise
- Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing (2/3)
- Replication Through Removable Media
- Supply Chain Compromise (0/3)
- Trusted Relationship
- Valid Accounts (0/4)

Execution
12 techniques

- Command and Scripting Interpreter (0/8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (0/2)
- Native API
- Scheduled Task/Job (0/7)
- Shared Modules
- Software Deployment Tools
- System Services (0/2)
- User Execution (1/3)
- Windows Management Instrumentation

Persistence
19 techniques

- Account Manipulation (1/4)
- BITS Jobs
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (0/3)
- Create or Modify System Process (0/4)
- Event Triggered Execution (2/15)
- External Remote Services
- Hijack Execution Flow (0/11)
- Implant Internal Image
- Modify Authentication Process (0/4)
- Office Application Startup (0/6)
- Pre-OS Boot (1/5)
- Scheduled Task/Job (0/7)
- Server Software Component (0/3)
- Traffic Signaling (1/1)
- Valid Accounts (0/4)

Privilege Escalation
13 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Create or Modify System Process (0/4)
- Domain Policy Modification (0/2)
- Escape to Host
- Event Triggered Execution (2/15)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (0/11)
- Process Injection (0/11)
- Scheduled Task/Job (0/7)
- Valid Accounts (0/4)

Defense Evasion
39 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- BITS Jobs
- Build Image on Host
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (0/2)
- Execution Guardrails (0/1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (0/2)
- Hide Artifacts (0/7)
- Hijack Execution Flow (0/11)
- Impair Defenses (0/7)
- Indicator Removal on Host (0/6)
- Indirect Command Execution
- Masquerading (0/6)
- Modify Authentication Process (0/4)
- Modify Cloud Compute Infrastructure (0/4)
- Modify Registry
- Modify System Image (0/2)
- Network Boundary Bridging (0/1)
- Obfuscated Files or Information (0/5)
- Pre-OS Boot (1/5)
- Process Injection (0/11)
- Rogue Domain Controller
- Rootkit
- Signed Binary Proxy Execution (1/11)
- Signed Script Proxy Execution (0/1)
- Subvert Trust Controls (0/6)
- Template Injection
- Traffic Signaling (1/1)
- Trusted Developer Utilities Proxy Execution (0/1)
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material (2/4)
- Valid Accounts (0/4)

Credential Access
15 techniques

- Brute Force (2/4)
- Credentials from Password Stores (0/5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials (0/2)
- Input Capture (0/4)
- Man-in-the-Middle (1/2)
- Modify Authentication Process (0/4)
- Network Sniffing
- OS Credential Dumping (1/8)
- Steal Application Access Token
- Steal or Forge Kerberos Tickets (1/4)
- Steal Web Session Cookie
- Two-Factor Authentication Interception
- Unsecured Credentials (0/7)

Discovery
27 techniques

- Account Discovery (0/4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Container and Resource Discovery
- Domain Trust Discovery
- File and Directory Discovery
- Network Service Scanning
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (0/3)
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery (0/1)
- System Information Discovery
- System Location Discovery
- System Network Configuration Discovery (0/1)
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion (0/3)

Lateral Movement
9 techniques

- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (0/2)
- Remote Services (2/6)
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material (2/4)

Collection
17 techniques

- Archive Collected Data (0/3)
- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Cloud Storage Object
- Data from Configuration Repository (0/2)
- Data from Information Repositories (0/2)
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged (0/2)
- Email Collection (0/3)
- Input Capture (0/4)
- Man in the Browser
- Man-in-the-Middle (1/2)
- Screen Capture
- Video Capture

Command and Control
16 techniques

- Application Layer Protocol (4/4)
- Communication Through Removable Media
- Data Encoding (0/2)
- Data Obfuscation (0/3)
- Dynamic Resolution (0/3)
- Encrypted Channel (2/2)
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- Non-Standard Port
- Protocol Tunneling
- Proxy (4/4)
- Remote Access Software
- Traffic Signaling (1/1)
- Web Service (0/3)

Exfiltration
9 techniques

- Automated Exfiltration (0/1)
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol (3/3)
- Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium (0/1)
- Exfiltration Over Physical Medium (0/1)
- Exfiltration Over Web Service (2/2)
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
13 techniques

- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation (1/3)
- Defacement (0/2)
- Disk Wipe (0/2)
- Endpoint Denial of Service (1/4)
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service (2/2)
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

MITRE Mapping Two - GSA

Reconnaissance
10 techniques

- Active Scanning (0/2)
- Gather Victim Host Information (0/4)
- Gather Victim Identity Information (0/3)
- Gather Victim Network Information (0/6)
- Gather Victim Org Information (0/4)
- Phishing for Information (0/3)
- Search Closed Sources (0/2)
- Search Open Technical Databases (0/5)
- Search Open Websites/Domains (0/2)
- Search Victim-Owned Websites

Resource Development
7 techniques

- Acquire Infrastructure (0/6)
- Compromise Accounts (0/2)
- Compromise Infrastructure (0/6)
- Develop Capabilities (0/4)
- Establish Accounts (0/2)
- Obtain Capabilities (0/6)
- Stage Capabilities (0/5)

Initial Access
9 techniques

- Drive-by Compromise
- Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing (2/3)
- Replication Through Removable Media
- Supply Chain Compromise (0/3)
- Trusted Relationship
- Valid Accounts (0/4)

Execution
12 techniques

- Command and Scripting Interpreter (0/8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (0/2)
- Native API
- Scheduled Task/Job (0/7)
- Shared Modules
- Software Deployment Tools
- System Services (0/2)
- User Execution (1/3)
- Windows Management Instrumentation

Persistence
19 techniques

- Account Manipulation (1/4)
- BITS Jobs
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (0/3)
- Create or Modify System Process (0/4)
- Event Triggered Execution (2/15)
- External Remote Services
- Hijack Execution Flow (0/11)
- Implant Internal Image
- Modify Authentication Process (0/4)
- Office Application Startup (0/6)
- Pre-OS Boot (1/5)
- Scheduled Task/Job (0/7)
- Server Software Component (0/3)
- Traffic Signaling (1/1)
- Valid Accounts (0/4)

Privilege Escalation
13 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- Boot or Logon Autostart Execution (0/14)
- Boot or Logon Initialization Scripts (0/5)
- Create or Modify System Process (0/4)
- Domain Policy Modification (0/2)
- Escape to Host
- Event Triggered Execution (2/15)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (0/11)
- Process Injection (0/11)
- Scheduled Task/Job (0/7)
- Valid Accounts (0/4)

Defense Evasion
39 techniques

- Abuse Elevation Control Mechanism (0/4)
- Access Token Manipulation (0/5)
- BITS Jobs
- Build Image on Host
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (0/2)
- Execution Guardrails (0/1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (0/2)
- Hide Artifacts (0/7)
- Hijack Execution Flow (0/11)
- Impair Defenses (0/7)
- Indicator Removal on Host (0/6)
- Indirect Command Execution
- Masquerading (0/6)
- Modify Authentication Process (0/4)
- Modify Cloud Compute Infrastructure (0/4)
- Modify Registry
- Modify System Image (0/2)
- Network Boundary Bridging (0/1)
- Obfuscated Files or Information (0/5)
- Pre-OS Boot (1/5)
- Process Injection (0/11)
- Rogue Domain Controller
- Rootkit
- Signed Binary Proxy Execution (1/11)
- Signed Script Proxy Execution (0/1)
- Subvert Trust Controls (0/6)
- Template Injection
- Traffic Signaling (1/1)
- Trusted Developer Utilities Proxy Execution (0/1)
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material (2/4)
- Valid Accounts (0/4)

Credential Access
15 techniques

- Brute Force (2/4)
- Credentials from Password Stores (0/5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials (0/2)
- Input Capture (0/4)
- Man-in-the-Middle (1/2)
- Modify Authentication Process (0/4)
- Network Sniffing
- OS Credential Dumping (1/8)
- Steal Application Access Token
- Steal or Forge Kerberos Tickets (1/4)
- Steal Web Session Cookie
- Two-Factor Authentication Interception
- Unsecured Credentials (0/7)

Discovery
27 techniques

- Account Discovery (0/4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Container and Resource Discovery
- Domain Trust Discovery
- File and Directory Discovery
- Network Service Scanning
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (0/3)
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery (0/1)
- System Information Discovery
- System Location Discovery
- System Network Configuration Discovery (0/1)
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion (0/3)

Lateral Movement
9 techniques

- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (0/2)
- Remote Services (2/6)
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material (2/4)

Collection
17 techniques

- Archive Collected Data (0/3)
- Audio Capture
- Automated Collection
- Clipboard Data
- Data from Cloud Storage Object
- Data from Configuration Repository (0/2)
- Data from Information Repositories (0/2)
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged (0/2)
- Email Collection (0/3)
- Input Capture (0/4)
- Man in the Browser
- Man-in-the-Middle (1/2)
- Screen Capture
- Video Capture

Command and Control
16 techniques

- Application Layer Protocol (4/4)
- Communication Through Removable Media
- Data Encoding (0/2)
- Data Obfuscation (0/3)
- Dynamic Resolution (0/3)
- Encrypted Channel (2/2)
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- Non-Standard Port
- Protocol Tunneling
- Proxy (4/4)
- Remote Access Software
- Traffic Signaling (1/1)
- Web Service (0/3)

Exfiltration
9 techniques

- Automated Exfiltration (0/1)
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol (3/3)
- Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium (0/1)
- Exfiltration Over Physical Medium (0/1)
- Exfiltration Over Web Service (2/2)
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
13 techniques

- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation (1/3)
- Defacement (0/2)
- Disk Wipe (0/2)
- Endpoint Denial of Service (1/4)
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service (2/2)
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

MITRE Mapping - Conclusion

- MDE and GSA map to roughly the same network related offensive techniques

- Mapping One – GSA has higher average confidence scores
  - ➔ More specific detection capabilities
  - ➔ TLS Inspection main reason for higher confidence mapping in C2 and Extfiltration

- Mapping Two – MDE has more defense capability mappings
  - ➔ MDE analyses more protocols than GSA (SMB, RDP, SSH, FTP, Etc.)
  - ➔ MDE analyses incoming traffic as well

What is next?

Improvements on my wish list

GSA Features

- Feature roadmap part of private previews
- I am most excited for
  - SSL/TLS Decryption
  - Alerting mechanism

Improved HTTP logging for GSA

- Especially needed when TLS Inspection is available
- MDE HttpConnectionInspected unreliable due to event capping

Expected HTTP Logging data	MDE - HttpConnectionInspected Columns	GSA - NetworkTrafficLogs Columns
Duration	Not available	Not available; Although we might be able to calculate it via ConnectionStatus column (not verified)
HTTP Status	AdditionalFields.status_code	ResponseCode
Bytes In	AdditionalFields.response_body_len	ReceivedBytes
Bytes Out	AdditionalFields.response_body_len	SentBytes
HTTP Method	AdditionalFields.method	HTTP Method
URL Hostname	AdditionalFields.host	DestinationFqdn
URL Path	AdditionalFields.uri	DestinationUrl
MIME(Content) type	Not available (only in FtpConnectionInspected)	Not available
User Agent	AdditionalFields.user_agent	Not available
URL Category	Not available	DestinationWebCategories
HTTP Version	AdditionalFields.version	Not available
Protocol	ActionType	TrafficType
File Name	Can be found via other MDE tables	Not available
Threat Type	Not available	ThreatType
Referrer Header	Not available	ReferrerHeader
XForwardedFor	Not available	XForwardedFor

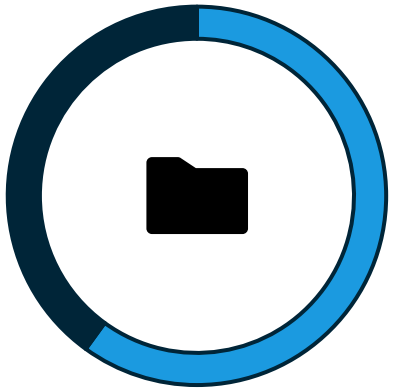
Closing

Let’s wrap this up

Closing

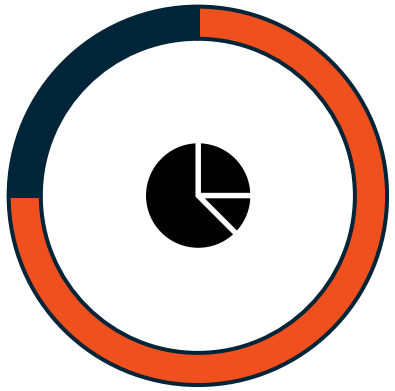
GSA and TLS

Once Global Secure Access supports TLS inspection, you will need it to have insights in encrypted traffic.



MDE Logging

MDE transactional logging logs more than HTTP traffic alone, while GSA is focused on HTTP/S traffic



MDE x GSA

For HTTP traffic, we have almost everything with MDE and GSA together (but since MDE capping prunes the logs this is unreliable). Process details from MDE and HTTP detail from GSA are super powerful together.

Alerting

GSA future alert mechanism will fill gaps the MDE alert mechanism is not able to cover.

Thank You