Microsoft Message analyzer Version 1.3

Known Issues and Workarounds

This document contains known issues and work arounds for Microsoft Message Analyzer v1.3. Browse this list to discover a solution to an issue you may be having. Otherwise, please report the problem to Microsoft.

# **Table of Contents**

[Capturing 3](#_Toc419807673)

[Network connections are reset when Message Analyzer is installed 3](#_Toc419807674)

[Web traffic can no longer be viewed and Internet Explorer is not working the same anymore 3](#_Toc419807675)

[Cannot start capturing or no data is being received 3](#_Toc419807676)

[Some messages seem to be missing after running a Live Trace Session 3](#_Toc419807677)

[You receive the error “Failed to start one or more trace session(s) due to the following error(s) “Live consumer xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx fails to start. Unable to start filter info provider service.” 4](#_Toc419807678)

[PowerShell capture trace is not saved to the PS execute path if you specify a relative path 4](#_Toc419807679)

[Simultaneous captures involving the same provider may have unpredictable results 4](#_Toc419807680)

[Provider settings in the Edit Session dialog may not apply to a new Live Trace session tab that uses the same provider as an existing Live Trace session tab 4](#_Toc419807681)

[Only a single ETW session can run on a single remote computer 4](#_Toc419807682)

[Loopback and Unencrypted IPSEC Trace Scenario does not work in some cases 5](#_Toc419807683)

[Identical multiple data sources with different filter configurations can result in captured message discrepancies 5](#_Toc419807684)

[Enumeration of network adapters fails when invalid username format is used 5](#_Toc419807685)

[The MinimumBufferCount and FlushTimer settings in ETW Session Configuration may fail to affect the ETW Session in which they are set 5](#_Toc419807686)

[The Pre-Encryption for HTTPS live trace scenario causes issues with Internet Explorer and Windows Store applications 5](#_Toc419807687)

[Information Disclosure on WebProxy Trace Scenario for multi user scenario 6](#_Toc419807688)

[Hyper-V traffic between virtual machines is not captured in Windows Server 2008 R2 6](#_Toc419807689)

[An error occurs when the Pre-Encryption for HTTPS trace scenario is running and a user attempts to send feedback 6](#_Toc419807690)

[Remote Capture 6](#_Toc419807691)

[WinRM service must be enabled on target computers in remote tracing scenarios 6](#_Toc419807692)

[Supported Remote Capture Scenarios 7](#_Toc419807693)

[Windows 8.1 and Windows Server 2012 R2 specific issues 7](#_Toc419807694)

[Capture on Local Link Layer fails (“Local Network Interfaces” on Windows 8.1 and later) 7](#_Toc419807695)

[UI 7](#_Toc419807696)

[Can't see columns for USB (or other) Events 7](#_Toc419807697)

[Cannot delete Chart data mappings 8](#_Toc419807698)

[Rogue Chart data mappings may degrade Message Analyzer performance 8](#_Toc419807699)

[A WebClientRequest exception can occur when auto-syncing all items with the Sync All Displayed Items button on the Downloads tab of the Asset Manager dialog 8](#_Toc419807700)

[Misleading error messages when invalid user credentials are used 8](#_Toc419807701)

[Message Analyzer crashes in various situations with System.IO.IOException “The file exists” when calling System.IO.Path.GetTempFileName() 8](#_Toc419807702)

[Performance 9](#_Toc419807703)

[Size of Traces that can be loaded/Number of Messages that can be captured 9](#_Toc419807704)

[Dropping Messages while Capturing 9](#_Toc419807705)

[Data Loading Rates Vary 9](#_Toc419807706)

[Errors on 32-bit machines 9](#_Toc419807707)

[Filtering 10](#_Toc419807708)

[IPv4 and IPv6Adress filters do not work on Wifi 10](#_Toc419807709)

[Why does MA capture duplicate TCP retransmits and diagnosis messages ? 10](#_Toc419807710)

[Pattern Expressions 10](#_Toc419807711)

[What are the Pattern Expression limitations? 10](#_Toc419807712)

[Opening Traces 12](#_Toc419807713)

[MA is unable to decode WPP-generated ETL traces 12](#_Toc419807714)

[Clicking multiple files from Windows Explorer doesn’t do anything 12](#_Toc419807715)

[Message Analyzer may not parse all saved files when loaded through the Open feature on the File menu 13](#_Toc419807716)

[Loading an matp file containing decrypted data displays the data as encrypted unless you choose to reparse 13](#_Toc419807717)

[Truncated Parsing enabled in Data Retrieval Sessions for .cap, .pcap, and .pcapng files only 13](#_Toc419807718)

[OPN Parsers 13](#_Toc419807719)

[Some earlier version OPN custom user parsers may not work as expected 13](#_Toc419807720)

[Incorrect parsing of Kerberos messages in relation to NDR and ASN decoding 13](#_Toc419807721)

[When opening an matp file containing Office data, Message Analyzer may not always reparse the data 14](#_Toc419807722)

# Capturing

## **Network connections are reset when Message Analyzer is installed**

Message Analyzer installs the Microsoft-PEF-NDIS driver on computers running the Windows 7, Windows 8, or Windows Server 2012 operating system. Currently, when this driver/provider is installed on a computer running the Windows 7 operating system, the network stack may reset. This might cause a temporary loss of network access, which can interfere with programs that rely on a network connection. Note that this problem has been mitigated on computers running Windows 8, Windows 2012, and later operating systems..

## **Web traffic can no longer be viewed and Internet Explorer is not working the same anymore**

Message Analyzer uses Fiddler to create a man-in-the-middle proxy to capture unencrypted web traffic. When the Message Analyzer closes unexpectedly, Message Analyzer tries to recover the original proxy settings; however, there are times when this may not occur. To fix this issue, try restarting and then stopping a Web Proxy capture OR resetting your proxy settings in the LAN settings section of the Connections Tab in Internet Options within Internet Explorer.

## **Cannot start capturing or no data is being received**

There is a limit to the number of capture sessions which can run concurrently. If the Message Analyzer isn't properly closed, these can accumulate and prevent new ones from running. To close these extra sessions:

1. Open the **Computer Management** utility by right-clicking *Computer* in the *Start Menu* and selecting *Manage*.
2. Open up the **Performance** tree category under **System Tools** and find the Event Trace Sessions folder under **Data Collector Sets**.
3. Find any session names that start with **Message Analyzer-<TraceScenarioName>** or **MMA-ETW-<GUID>**; right-click and stop them.
4. Then right-click them again and **Delete** them.

## **Some messages seem to be missing after running a Live Trace Session**

If event throughput is very high, Message Analyzer might be dropping messages. Microsoft suggests that you use a **Fast Filter** to prevent this from happening.

**Workaround** To resolve this issue, do the following:

1. Select **File** | **New Session** | **Live Trace**
2. In the **Select a trace scenario** drop-down list on the **Live Trace** tab, select a **Trace Scenario** that has a provider that uses **Fast Filters**.

For example, to capture DNS messages across a firewall, select the **Loopback and Unencrypted IPSEC** scenario.

1. In the **ETW Provider**s list, click the **Configure** link to display the **Advanced Settings** dialog for the applicable message provider.
2. In the dialog, click the **Provider** tab and then in the **Fast Filters** pane select **UDPPort** in the **Fast Filter 1** drop-down list.
3. In the text box to the right of the **Fast Filter** drop-down list, specify the port number 53.
4. Click **OK** to exit the **Advanced Settings** dialog and then start your Live Trace Session in the **New Session** dialog by clicking the **Start** button.

## **You receive the error “Failed to start one or more trace session(s) due to the following error(s) “Live consumer xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx fails to start. Unable to start filter info provider service.”**

This happens when you start a **Trace Scenario** that uses the **Microsoft-PEF-WFP-MessageProvider** without running Message Analyzer as administrator.

**Workaround** To resolve this issue, save your work, exit Message Analyzer, and then do the following:

1. Go to the Command Prompt.
2. At the command line, type “sc stop wfpcapture” to stop the PEF WFP driver.
3. Restart Message Analyzer by right-clicking "Microsoft Message Analyzer" on the **Start** menu or task bar of your computer and then select **"Run as administrator"**.

## **PowerShell capture trace is not saved to the PS execute path if you specify a relative path**

When running a PowerShell script as Admin, the path variable for Current Directory is set to the System32 directory. So any relative path trace files will but created in System32. To work around the issue, specify a fully qualified path starting at the drive when capturing with PowerShell as Admin.

## **Simultaneous captures involving the same provider may have unpredictable results**

Starting simultaneous captures involving the same provider is not recommended. It is not possible to configure different instances of the same provider and attempting to start multiple instances of the same provider can provide unpredictable results.

## **Provider settings in the Edit Session dialog may not apply to a new Live Trace session tab that uses the same provider as an existing Live Trace session tab**

Because a Live Trace Session will hold the created plugin instance until its associated Live Trace session tab is closed, session reconfiguration changes related to provider settings may not apply, for example **Fast Filters**. This restriction applies to all Live Trace Sessions that contain a provider with a plugin instance, including the following:

* Microsoft-Windows-NDIS-PacketCapture
* Microsoft-PEF-Windows-PacketCapture
* Microsoft-Pef-WFP-MessageProvider
* Microsoft-Pef-WebProxy

For example, if you modify a **Fast Filter** for the **Microsoft-PEF-NDIS-PacketCapture** provider without closing any previous Live Trace session tab that uses the same provider, then the modified filters will not apply.

## **Only a single ETW session can run on a single remote computer**

Message Analyzer can have only one ETW session running at any time on a single remote host. For example, you cannot have one session that uses the **Microsoft- Windows-NDIS-PacketCapture** provider and another session that uses a different provider both running on the same remote computer at the same time. For another ETW session to run on the same host, any previous session must be closed, due to WMI restrictions.

## **Loopback and Unencrypted IPSEC Trace Scenario does not work in some cases**

The **Loopback and Unencrypted IPSEC** Trace Scenario does not work in the following user scenarios:

* Using multiple instances of Message Analyzer to capture live data with this Trace Scenario.
* Multiple users logged onto the server machine and simultaneously capturing live data with this Trace Scenario.

## **Identical multiple data sources with different filter configurations can result in captured message discrepancies**

Message Analyzer currently allows you to add the same data source multiple times to a Live Trace Session or a Data Retrieval Session by using the **New Data Source** tab in the **New Session** dialog. This can cause discrepancies in the messages that are captured, for example, when one configuration of the data source has a **Fast Filter** or **Session Filter** applied and the other does not. Also when attempting to stop such a session with multiple data sources, a ***Stop Trace Error*** can occur.

## **Enumeration of network adapters fails when invalid username format is used**

In remote tracing scenarios that require a domain configuration, the username entry should be in the form: *domain\username*.

Otherwise, if only the username is specified without the domain name, enumeration of network adapters fails.

## **The MinimumBufferCount and FlushTimer settings in ETW Session Configuration may fail to affect the ETW Session in which they are set**

Values set in the **ETW Session - Advanced Configuration** dialog for **MinimumBufferCount** and **FlushTimer** for Live Trace Sessions with the **Microsoft-Windows-NDIS-PacketCapture** provider are not transferred to the ETW Session, rather, the default values persist.

## **The Pre-Encryption for HTTPS live trace scenario causes issues with Internet Explorer and Windows Store applications**

You might find when you try to run the **Pre-Encyption for HTTPS** **Trace Scenario** with the **WebProxy** provider that the application you are tracing fails to work or that Message Analyzer doesn’t capture any traffic.

This happens because Windows now protects client-to-client traffic by disabling local loopback to 127.0.0.1 in certain conditions. This interferes with the way that **WebProxy** captures traffic.

Windows 8 has EPM (Enhanced Protected Mode) enabled default for the Windows 8 Internet Explorer Application (the desktop version is not enabled). This mode includes the option to block EMP. You can either remove this option, or change the Loopback exemption directly by using the information below.

Windows 8.1 client and server have EPM enabled by default at this time for both versions of IE.

Windows 8 and 8.1 have the loopback option disabled for all Windows Store applications. You have to use one of the workarounds below to enable tracing for a specific Windows Store application.

**Workaround(s)**

1. If the Web client is IE 10, then Enhanced Protected Mode has to be unchecked in the advanced settings or on Windows 8 or later execute the command **"CheckNetIsolation.exe loopbackExempt -a -n=Windows*ie*ac\_001"** to enable the loopback exemption for IE.
2. On Windows 8 or later, if the Webclient is store app, then following command has to be executed **"CheckNetIsolation.exe loopbackExempt -a -n=<Appcontainer name of the Web client application>"** to enable the loopback exemption for Windows Store applications.

Reference <http://msdn.microsoft.com/en-us/library/windows/apps/Hh780593.aspx>.
Capturing with the **WebProxy** provider uses the Fiddler core API which has some known limitations and issues:

* Untrusted certificate with SSL capturing won't decode.
* Proxy settings not reverted when MMA crashes on capturing with WebProxy.
* The **Unencypted HTTPS** **Trace Scenario** with the **WebProxy** provider won't work in cases such as Azure, where you need a dedicated certificate instead of the fake Fiddler certificate.
* Cannot capture a site that requires additional authentication, for instance Channel binding tokens.
* There's no way to know the actual process ID or name of the traffic from the **WebProxy** provider.
* Cannot capture traffic which does not use proxy settings as set in Internet Options for Internet Explorer.

## **Information Disclosure on WebProxy Trace Scenario for multi user scenario**

If an administrator adds the two users User1 and User2 to the Message Capture Users group and both users are remotely logged in at the same time, User1 can see the traffic of User2 and vice versa, while using MA. The reason is that an ETW session is global. Further, it is assumed that these added users will have capture capabilities at the system level in such cases.

## **Hyper-V traffic between virtual machines is not captured in Windows Server 2008 R2**

On Windows Server 2008 R2 Hyper-V traffic is only captured between the host and any virtual machine. Traffic from a virtual machine targeted to another virtual machine is not captured.

## **An error occurs when the Pre-Encryption for HTTPS trace scenario is running and a user attempts to send feedback**

Due to SSL interference, the **Pre-Encryption for HTTPS Trace Scenario** that uses the **WebProxy-Fiddler** provider can block the Feedback process. When this occurs, the following **Error Submitting Feedback** message is displayed:

"An error occurred while submitting feedback."

**Workaround** To avoid this issue, stop your network capture before you submit Feedback.

# Remote Capture

## **WinRM service must be enabled on target computers in remote tracing scenarios**

In remote tracing scenarios, the ‘winrm’ service must be enabled on remote target computers. The current generic **Invalid Host Name** error message does not sufficiently indicate this issue.

## **Supported Remote Capture Scenarios**

Supported servers (remote capture target):

* Windows Server 2012 R2

Supported clients (remote capture source):

* Windows 7 (needs WMF 3.0 <http://www.microsoft.com/en-pk/download/details.aspx?id=34595>)
* Windows Server 2008 R2 (needs WMF 3.0 <http://www.microsoft.com/en-pk/download/details.aspx?id=34595>)
* Windows 8
* Windows Server 2012
* Windows Blue 8.1 (build 9600)
* Windows Server 2012 R2 (build 9600)

The following are the supported capture scenarios:

* Both client and server being domain-joined.
* Both client and server being in workgroup.
* When the Client is domain-joined and server is in workgroup.
* The last case (client in workgroup and server is domain-joined) is supported, but IPSec needs to be disabled on the server; therefore, this is not a recommended scenario.
* Client and Server are in different trusted domains in same forest.

Special considerations:

* If credentials are not provided, the current logged on user’s credentials (on the client) are used for establishing a connection to server.
* When the client is domain-joined and the server is in workgroup, the remote machine needs to be added to the trusted hosts list on the client by running the following commands from PowerShell or an elevated command prompt:

WinRM quickconfig -quiet

WinRM --% set winrm/config/client @{TrustedHosts="RemoteHostName"}

# Windows 8.1 and Windows Server 2012 R2 specific issues

## **Capture on Local Link Layer fails (“Local Network Interfaces” on Windows 8.1 and later)**

User needs to ***always run as administrator*** to capture message data with the **Local Network Interfaces** **Trace Scenario** on Windows 8.1 Client and Windows Server 2012 R2 machines.

# **UI**

## **Can't see columns for USB (or other) Events**

Some fields for providers cannot be seen until they are loaded for the first time. USB and other provider parsers are created dynamically the first time you open or start a new trace for that provider. You will be unable to see the provider fields in the **Column Chooser** **Tool Window** nor can you use them for filtering until the parser is created. Once the parser is created, fields that you add as columns are preserved, even if you reset the parser by removing it manually.

## **Cannot delete Chart data mappings**

Data mappings for charts cannot be removed from the UI.

## **Rogue Chart data mappings may degrade Message Analyzer performance**

Charts can leave rogue data mappings that could affect performance. You should manually remove any unused maps before publishing assets, by editing the specific asset file.

**Workaround** You can edit the XML reference which starts with <DataCollector> if you must remove the mapping. Export your assets from the library management system, make a change, and then re-import the asset.

## **A WebClientRequest exception can occur when auto-syncing all items with the Sync All Displayed Items button on the Downloads tab of the Asset Manager dialog**

When selecting the **Sync All Displayed Items** on the **Downloads** tab of the Message Analyzer **Asset Manager** dialog to sync multiple items, a ***WebClient Request*** exception may be reported for unsuccessful downloads of some asset packages and collections.

**Workaround**  If this occurs, Auto-Sync each asset package or collection manually.

## **Misleading error messages when invalid user credentials are used**

When applying invalid username and passwords to a Live Trace Session on the local host, the resulting ***Capture Session Error*** and ***Start Trace Error*** messages can be misleading.

For example, after stopping a live session following the ***Capture Session Error*** message, and then restarting, the ***Start Trace Error*** message can imply that the username and password are valid when they are not.

## **Message Analyzer crashes in various situations with System.IO.IOException “The file exists” when calling System.IO.Path.GetTempFileName()**

Message Analyzer may crash when attempting to create a temporary file using System.IO.Path.GetTempFileName(), but more than 65535 files have been created previously and not deleted. Any application using this method may leave temporary files behind and contribute to this problem. Below are known cases when Message Analyzer may crash, but other components that Message Analyzer uses may also attempt to create a temporary file and result in a crash.

* Right-click in the **Field Data** tool window and select the “Open in Default Application” menu
* Click on the **Diagnostics** icon for a message in the **Analysis Grid** to see details

**Workaround** Open the %TEMP% folder in File Explorer and delete any unneeded files.

# Performance

## **Size of Traces that can be loaded/Number of Messages that can be captured**

The number of messages that can be captured or the size of trace file that can be loaded is dependent on the amount total memory (actual + virtual using paging file) on the machine.   Paging file settings can be adjusted using the Control Panel | System applet.

## **Dropping Messages while Capturing**

By default, Message Analyzer has a 200MB queue in memory to store messages temporarily while messages are being processed.  If messages are incoming at a very fast rate, this queue can get filled and messages may be dropped.  If you suspect that Message Analyzer is dropping messages, you can adjust this queue by changing the **Live Trace Message Buffer** **Size** option in the **Options** dialog that is accessible from the Message Analyzer **File** menu.  Note that Message Analyzer can provide an indication of dropped packets in **Trace Scenarios** that use the **Microsoft-PEF-WFP-MessageProvider** is you select the **Select Discarded Packet Events** option in the **Advanced Setting** dialog for this provider.

Another way to avoid dropped packets is to use a **Fast Filter** to filter out messages at the driver level.  You can configure **Fast Filters** in **Trace Scenarios** that use the **Microsoft-PEF-WFP-MessageProvider** or the **Microsoft-PEF-NDIS-PacketCapture** provider.

## **Data Loading Rates Vary**

When you load non-matp traces into Message Analyzer, they will be parsed.   The following is an approximation of the current data loading rates for .cap files:

#### **.CAP files:**

* ~4,900 messages/second (up from ~2500) avg
* ~3 MB/second (up from ~2 MB/second) avg

#### **.MATP files:**

Though .matp files are already parsed and are native, you can reparse them through a Data Retrieval Session (**File** menu **>** **New Session** **>** **Files** item).  You might do this if you want to combine an .matp file with other traces or logs, so you can view the data as a single message set. In this case, the loading time will be similar to that of .cap files, as previously indicated.

However, when opening a .matp file by using the **Open** feature (**File**>**Open**>**From File Explorer**), double-clicking the file in Windows file **Explorer**, or by dragging and dropping it into Message Analyzer, the loading rate is significantly faster, as the messages are already parsed:

* ~10,000 messages/second avg
* ~10 MB /second avg

## **Errors on 32-bit machines**

On a 32 bit machine or a machine with limited memory, you may get random run-time errors, popups about insufficient memory, sudden exits, and halted parsing. This can happen when you parse a trace file that involves large amount of state information. In particular TCP connections, (around 10,000 connections) can cause this problem. In general, this problem is data dependent.

Each piece of state can remain in memory until the state is released. For instance, a TCP connection will introduce a separate data structure for parsing, which is not released dynamically until the end of parsing. When many connections need to be parsed simultaneously, the memory will be exhausted.

**Workaround** Use a machine that has more memory to parse such traces; ideally a 64 bit machine with a minimum of 8Gb memory.

# Filtering

## **IPv4 and IPv6Adress filters do not work on Wifi**

The IPv4 and IPv6 **LinkLevelAddress** **Fast Filters** fail to work on WiFi on Windows 7 64bit. No traffic will match these filters.

## **Why does MA capture duplicate TCP retransmits and diagnosis messages ?**

In **Trace Scenarios** that use the **Microsoft-PEF-WFP-MessageProvider**, Message Analyzer will capture duplicate traffic such as duplicate TCP Retransmits, unless you specifically filter out loopback traffic.

**Workaround** In the **Advanced Settings** dialog for the **Microsoft-PEF-WFP-MessageProvider**, configure these two **Fast Filters**: IPv4 != 127.0.0.1 and IPv6 !=::1, to block loopback traffic from being captured in your Live Trace Session so you can avoid erroneous TCP retransmits and other false diagnoses messages.

Note that the **Local Loopback Network** **Trace Scenario** with the **Microsoft-PEF-WFP-MessageProvider** specifically captures loopback traffic with the following **Fast Filters**: IPv4 = 127.0.0.1 and IPv6 =::1.

# Pattern Expressions

## **What are the Pattern Expression limitations?**

### **The ‘in’ parameter for creating collection is not supported**:

scenario S[out array<int> ids] = Request{ID in ids} interleave;

### **The Permute (&) operator is not supported**:

Won’t be compiled:

scenario S = A & B;

or scenario S = A permute B;

### **The ‘fork’ operator can only be the top most operator**:

Following definitions are not allowed:

scenario S = A | B -> (C || D);

scenario S = A || B || C;

### **Explicitly specify the type of out parameter in not allowed**:

Won’t be compiled:

scenario s[out binary payload] = HTTP.HttpContract.Operation{Payload is payload:binary};

Supported syntax:

scenario s[out binary payload] = HTTP.HttpContract.Operation{ Payload is payload };

### **Explicitly specifying the ‘in’ keyword for parameters is not allowed**:

Won’t be compiled:

scenario S[in string name] = HTTP.HttpContract.Operation{Method == name};

Supported syntax:

scenario S[string name] = HTTP.HttpContract.Operation{Method == name};

### **The “where” clause is not supported in virtual operation**:

Won’t be compiled:

virtual operation VOp

{

}

= MyScenario[out var reqId, out var statusCode]

where (StatusCode != 200) ==> !Success;

scenario MyScenario[out int reqId, out int statusCode] =

 accepts Request{ID is reqId}

 accepts Response{ID == reqId, StatusCode is statusCode};

### **The “exception” clause is not supported in virtual operation**:

Won’t be compiled:

virtual operation VOp

{

}

exception optional int = reason

=

accepts Request{ID is reqId:int}

(

 accepts Response{ID == reqId, StatusCode == 200, StatusCode is statusCode:int}

 |

 accepts Response{ID == reqId, StatusCode != 200, StatusCode is reason:int}

);

### **Referencing one scenario from another is not supported**:

Won’t be compiled:

scenario S1 = Relay{ID is var id} Relay{ID == id};

scenario S2 = Request{ID is var id} -> S1 -> Response{ID == id};

Please note: In MA’s Pattern Match View, it is not allowed to declare more than one scenario.

### **Limited support for referencing scenarios in virtual operations**:

Supported:

virtual operation M { … } = S(…);

scenario S(…) = …

Not Supported:

virtual operation M { … } = S1(…) -> S2(…)

### **In the case of overlapping matches, there is no guarantee that the longer one will be reported**:

scenario S = Request{ID is var reqId} -> Relay{ID == reqId}? -> Response{ID == reqId};

The input sequence is:

Request{ ID == 1 }

Request { ID == 2 }

Relay { ID == 1 }

Response { ID == 2 }

Response { ID == 1 }

Expected:

Request{ ID == 1 } Relay { ID == 1 } Response { ID == 1 }

Actual:

{{Request { ID == 2 } Response { ID == 2 }

# Opening Traces

## **MA is unable to decode WPP-generated ETL traces**

WPP traces require a configuration file as described in [Processing WPP-Generated Events](http://technet.microsoft.com/en-us/library/dn727250.aspx).

## **Clicking multiple files from Windows Explorer doesn’t do anything**

Selecting multiple files in the Windows file **Explorer** and selecting "Open with Message Analyzer" does not launch Message Analyzer. This is currently not supported.

**Workaround** Select a single file and select "Open with Message Analyzer". An alternative for opening multiple files is to launch Message Analyzer first, go to the **File** menu | **New Session** | **Files** and then add the files to a Data Retrieval Session.

## **Message Analyzer may not parse all saved files when loaded through the Open feature on the File menu**

When opening .matp files with the **Open** feature from the Message Analyzer **File** menu, or when double-clicking it, Message Analyzer may not fully parse the file. This can occur if the current OPN parser set is new compared to the OPN parser set used in the original capture that is saved to an .matp file. This can also occur if an .matp file contains an applied **Parsing Level**.

**Workaround**  Use a Data Retrieval Session (**File** button in the **New Session** dialog) to open the file, which always forces a reparse of matp files under these circumstances.

## **Loading an matp file containing decrypted data displays the data as encrypted unless you choose to reparse**

From the latest version of Message Analyzer, if you try to load a saved matp file that was originally captured with out-of-date OPN parsers in a previous version of Message Analyzer and you also decrypted the data with Message Analyzer, the data will display as encrypted unless you choose to reparse with up-to-date OPN parsers.

**Workaround** Always use the same Message Analyzer version for parsing that was used for saving a decrypted matp file. If you did not and you have recently installed the latest version of Message Analyzer, add the encryption certificate again in the new version of Message Analyzer so that you can decrypt as you reparse. Also be sure to choose the reparse option when you are prompted during data loading.

## **Truncated Parsing enabled in Data Retrieval Sessions for .cap, .pcap, and .pcapng files only**

The **Truncated Parsing** checkbox does not change the state of truncated parsing even if the checkbox is selected in the **New Session** or **Edit Session** dialog, *except* for files in .cap, .pcap, or .pcapng format.

# OPN Parsers

## **Some earlier version OPN custom user parsers may not work as expected**

In the Message Analyzer v1.2 release, backward compatibility with custom OPN Windows parsers is not supported due to compilation issues caused by technology and OPN changes. As a result, some user-written parsers may not work as expected.

## **Incorrect parsing of Kerberos messages in relation to NDR and ASN decoding**

Message Analyzer does not correctly parse Kerberos messages in relation to NDR and ASN decoding. For example, in an NDR conformant array, the max count value and padding are hidden and only the elements and associated binary in the HEX view of the Message Data window are shown.  In the case of ASN, the ASN tag and length are hidden and only the value and associated binary are shown.

## **When opening an matp file containing Office data, Message Analyzer may not always reparse the data**

This scenario consists of the following: If you capture Office traffic without any Office parsers installed in Message Analyzer, you will capture lower level network traffic only, as no parsing will occur. You then save the trace in the .matp file format. Thereafter, you install the latest Office OPN parsers through the Message Analyzer Sharing Infrastructure.

If you now load the saved matp file into Message Analyzer through the **Open** feature, the data is not reparsed.

**Workaround** Load the previously saved matp file into Message Analyzer through a Data Retrieval Session. Note that if you have the latest version of Message Analyzer installed before capturing Office traffic, you should never have an issue because the latest MOD (Office) parser set will be automatically downloaded to your Message Analyzer installation from the Microsoft web service that drives the **Message Analyzer** feed in the sharing infrastructure.