Malcolm Network Traffic Analysis Quick Start Guide

2021-04-12

Table of Contents

Query Syntax

Kibana supports two query syntaxes: the legacy Lucene syntax and the new Kibana Query Language (KQL), both of which are somewhat different than Arkime's query syntax (see the help by clicking on the   icon in the upper-left hand corner of the Arkime interface). The Arkime interface is for searching and visualizing both Arkime sessions and Zeek logs. The prebuilt dashboards in the Kibana interface are for searching and visualizing Zeek logs, but will not include Arkime sessions. Here are some common patterns used in building search query strings for Arkime and Kibana, respectively.

	Arkime Search String	**Kibana Search String (Lucene)**	**Kibana Search String (KQL)**
Field exists	`zeek.logType == EXISTS!`	`_exists_:zeek.logType`	`zeek.logType:*`
Field does not exist	`zeek.logType != EXISTS!`	`NOT _exists_:zeek.logType`	`NOT zeek.logType:*`
Field matches a value	`port.dst == 22`	`dstPort:22`	`dstPort:22`
Field does not match a value	`port.dst != 22`	`NOT dstPort:22`	`NOT dstPort:22`
Field matches at least one of a list of values	`tags == [external_source, external_destination]`	`tags:(external_source OR external_destination)`	`tags:(external_source or external_destination)`
Field range (inclusive)	`http.statuscode >= 200 && http.statuscode <= 300`	`http.statuscode:[200 TO 300]`	`http.statuscode >= 200 and http.statuscode <= 300`
Field range (exclusive)	`http.statuscode > 200 && http.statuscode < 300`	`http.statuscode:{200 TO 300}`	`http.statuscode > 200 and http.statuscode < 300`
Field range (mixed exclusivity)	`http.statuscode >= 200 && http.statuscode < 300`	`http.statuscode:[200 TO 300}`	`http.statuscode >= 200 and http.statuscode < 300`
Match all search terms (AND)	`(tags == [external_source, external_destination]) && (http.statuscode == 401)`	`tags:(external_source OR external_destination) AND http.statuscode:401`	`tags:(external_source or external_destination) and http.statuscode:401`
Match any search terms (OR)	`(zeek_ftp.password == EXISTS!) || (zeek_http.password == EXISTS!) || (zeek.user == "anonymous")`	`_exists_:zeek_ftp.password OR _exists_:zeek_http.password OR zeek.user:"anonymous"`	`zeek_ftp.password:* or zeek_http.password:* or zeek.user:"anonymous"`
Global string search (anywhere in the document)	all Arkime search expressions are field-based	`microsoft`	`microsoft`
Wildcards	`host.dns == "*micro?oft*"` (? for single character, * for any characters)	`dns.host:*micro?oft*` (? for single character, * for any characters)	`dns.host:*micro*ft*` (* for any characters)
Regex	`host.http == /.*www\.f.*k\.com.*/`	`zeek_http.host:/.*www\.f.*k\.com.*/`	Kibana Query Language does not currently support regex
IPv4 values	`ip == 0.0.0.0/0`	`srcIp:"0.0.0.0/0" OR dstIp:"0.0.0.0/0"`	`srcIp:"0.0.0.0/0" OR dstIp:"0.0.0.0/0"`
IPv6 values	`(ip.src == EXISTS! || ip.dst == EXISTS!) && (ip != 0.0.0.0/0)`	`(_exists_:srcIp AND NOT srcIp:"0.0.0.0/0") OR (_exists_:dstIp AND NOT`	`(srcIp:* and not srcIp:"0.0.0.0/0") or (dstIp:* and not dstIp:"0.0.0.0/0")`

|	**Arkime Search String**	**Kibana Search String (Lucene)**	**Kibana Search String (KQL)**
		dstIp:"0.0.0.0/0")	
GeoIP information available	country == EXISTS!	_exists_:zeek.destination_geo OR _exists_:zeek.source_geo	zeek.destination_geo:* or zeek.source_geo:*
Zeek log type	zeek.logType == notice	zeek.logType:notice	zeek.logType:notice
IP CIDR Subnets	ip.src == 172.16.0.0/12	srcIp:"172.16.0.0/12"	srcIp:"172.16.0.0/12"
Search time frame	Use Arkime time bounding controls under the search bar	Use Kibana time range controls in the upper right-hand corner	Use Kibana time range controls in the upper right-hand corner

Arkime

The Arkime interface is for searching and visualizing both Arkime sessions (with PCAP payloads) and Zeek logs.

Search bar

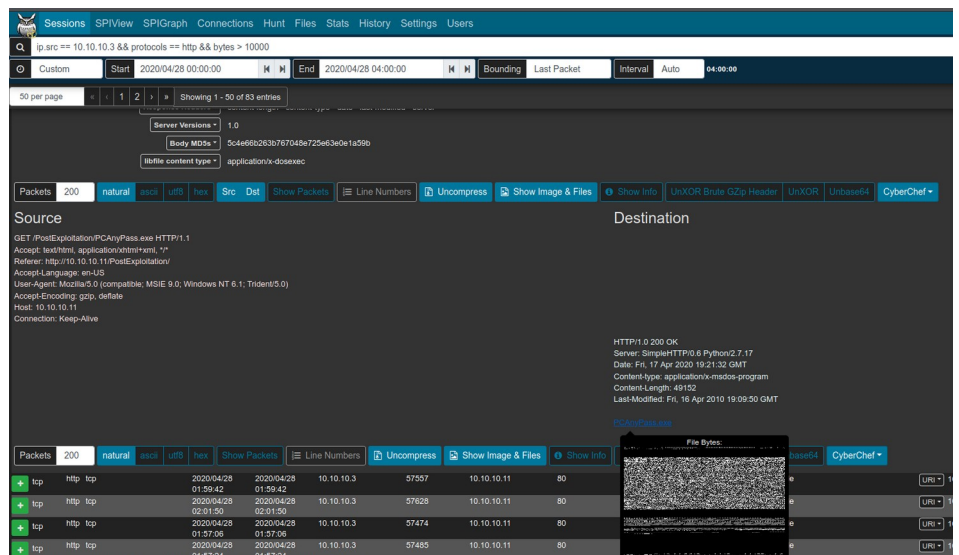Views

Time frame

Session details/payload
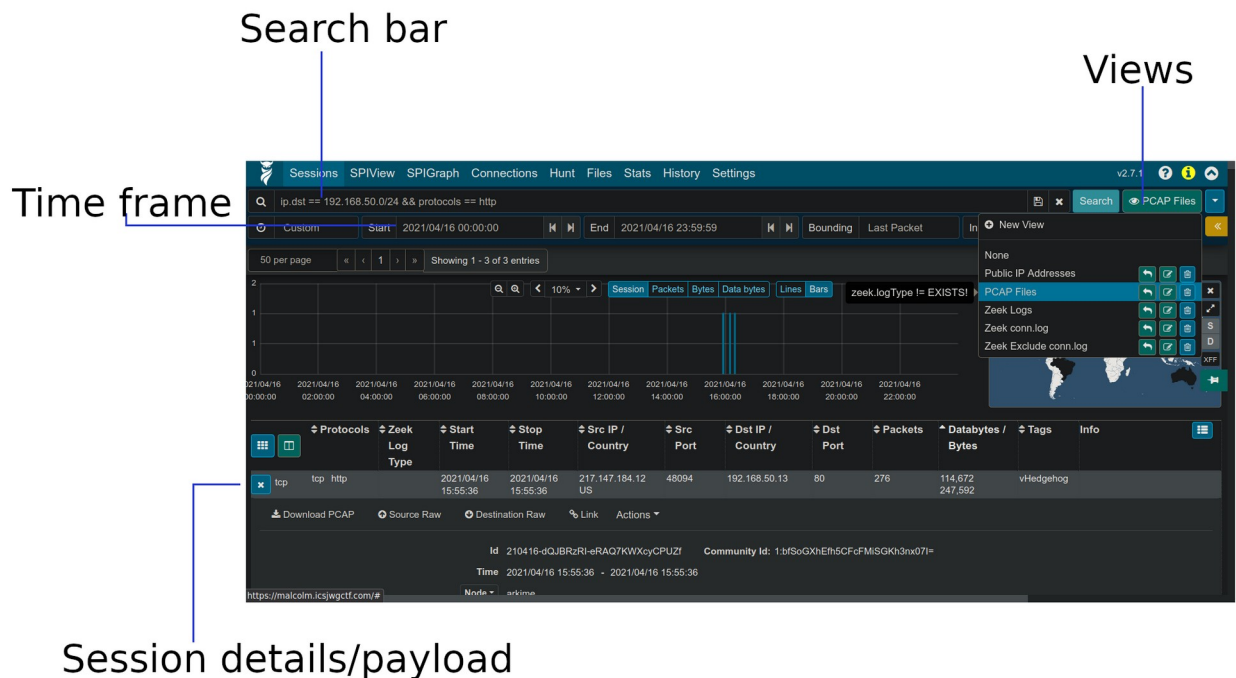
Sessions

Arkime's Sessions view is for viewing Arkime sessions and/or Zeek logs at the finest granularity: one line in the sessions table represents a single Arkime session or line from a Zeek log. It provides field-level details of sessions and logs matching the filters you define.

Arkime sessions are distinguished from Zeek logs by the absence of a value in the *Zeek Log Type* column.

Accessible to the right of the search bar via the   icon, views can be applied to filter between Arkime sessions (for which packet payloads are available through associated PCAP files) and Zeek logs.

Packet payloads can be viewed by expanding an Arkime session entry in the sessions table.

See the Malcolm documentation for techniques for to correlate Arkime sessions and Zeek logs using the community ID field and Zeek's connection UID.

SPIView

Arkime's SPIView lets you explore "top *n*" values and field cardinality for all fields of both Arkime sessions and Zeek logs. You can narrow in on items of interest by apply filters then pivot to the sessions or SPIGraph views for to view details or added context. Because SPIView runs many queries, make sure to limit the search time frame to ≤ 1 week before using it.
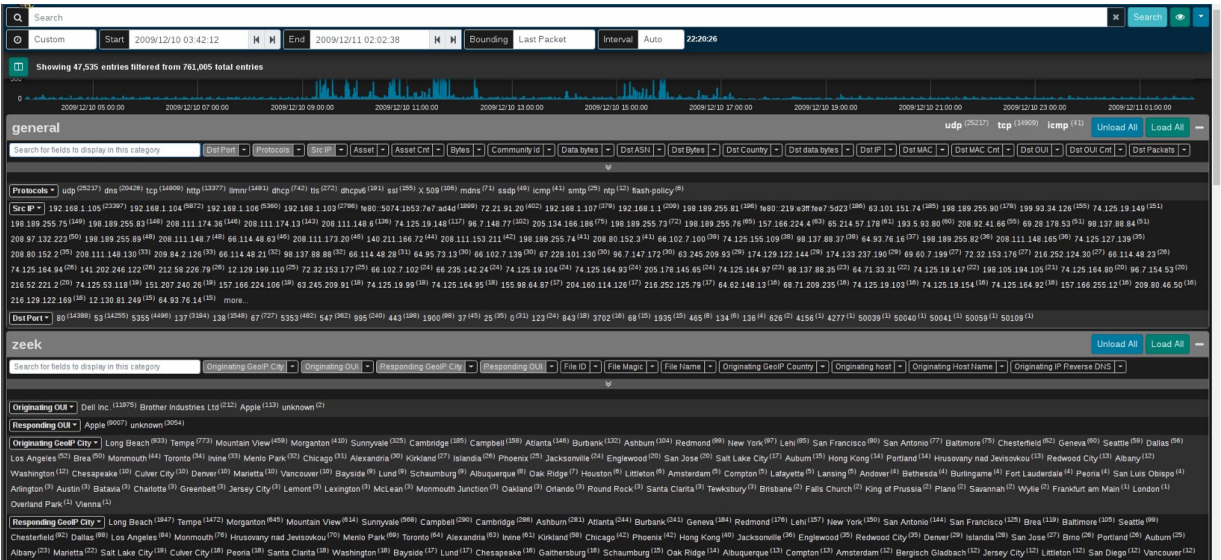

SPIGraph

SPIGraph helps you visualize "top *n*" field values chronologically and geographically, making it easier to identify trends and patterns in network traffic.

Connections

Using the Connections view you can visualize logical relationships between hosts. The graph's source and destination nodes can be configured to represent any combination of fields populated in Arkime sessions and Zeek logs.
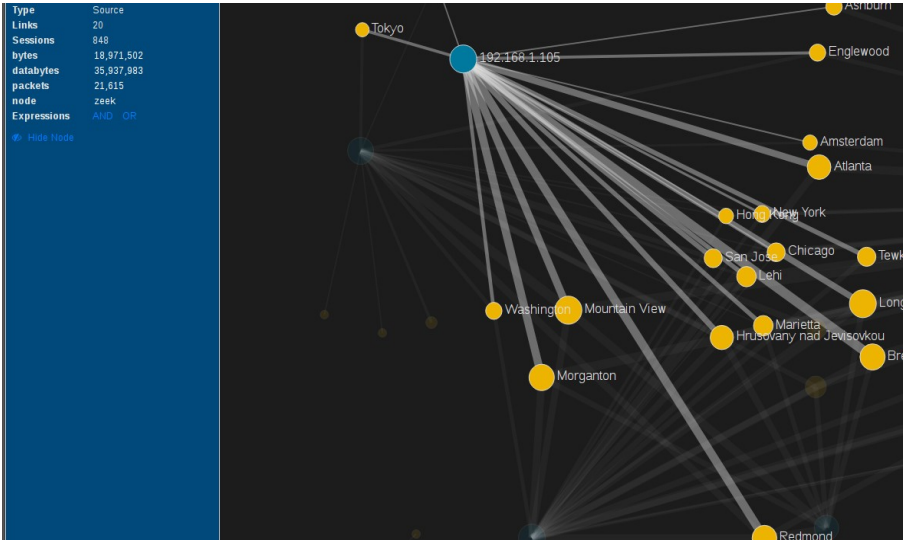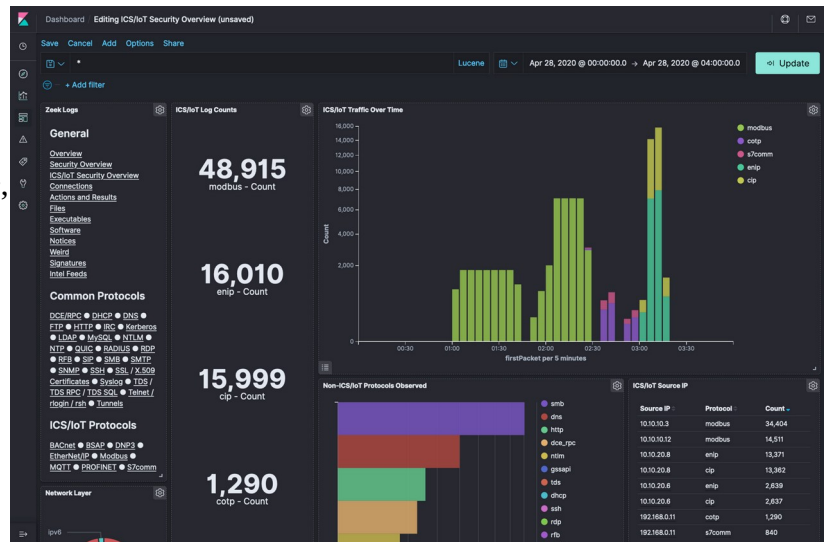
Kibana

Kibana provides intuitive interactive representations of Zeek log data that simplify the process of recognizing and narrowing in on important network events: starting from a high-level overview and being able to quickly "drill-down" to the traffic of an individual host or connection of interest.

Where possible, Malcolm correlates common fields from across different protocols to allow you to view one device's or application's network traffic in the context of the other traffic occurring around it. A good example of this is the *Actions and Results* dashboard, in which actions (such as "a file was written," "a logon was attempted," "a web page was requested") and results ("success," "access denied," "page not found") can be inspected together regardless of protocol.

Dashboards

Malcolm comes with dozens of prebuilt visualizations specifically for data ingested from Zeek logs. Its dashboards fall into two categories: overview dashboards (e.g., *Connections*, *Software*, *Files* and *Action and Results*) and protocol-specific dashboards (e.g., *HTTP*, *FTP*, *SMB* and *BACnet*).

See the Malcolm documentation for the list of protocols Malcolm parses from network traffic.

Discover

The Discover view enables you to view Zeek logs on a record-by-record basis, similar to Arkime's Sessions view.
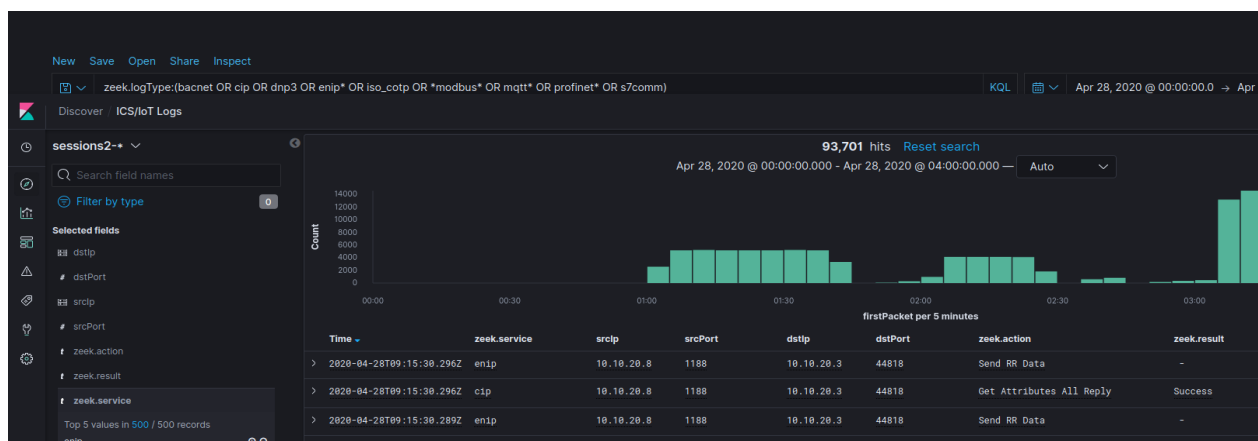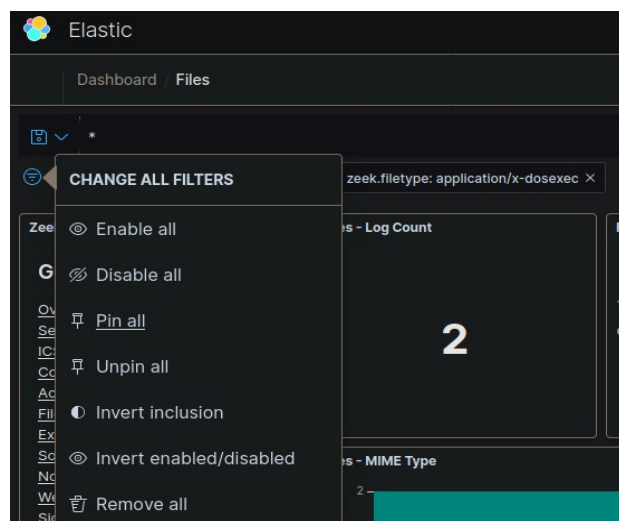
Filters

Kibana's query bar allows you to specify search constraints, using Lucene or KQL query syntax. Modifying the contents of this bar and hitting Enter or clicking the Search icon to the right will run the search and update the results displayed. See the *Query Syntax* table or the [Malcolm documentation](#) for examples of Kibana query syntax.

The filter bar is another way of specifying search constraints through a GUI. In most cases there's not really a meaningful distinction between putting query terms in via the query bar vs. the filter bar, although using the filter bar does allow you to more easily pin filters across different dashboards and is somewhat more intuitive.

Filters may also be populated by clicking on values in charts and graphs and choosing the magnifying glass icon with either the plus sign (+) or minus sign (-) to restrict to or exclude that value from the result set.

To preserve filters when moving between Kibana dashboards, use filter pinning. Click the *Change all filters* icon to the far left of the filter bar, then click *Pin all*.