



conn.logIP, TCP, UDP and ICMP connection details

Field	Туре	Description
ts	time	Timestamp of the first packet
uid	string	Unique ID of the connection
id.orig_h	addr	Originating endpoint's IP address (AKA Orig)
id.orig_p	port	Originating endpoint's TCP/UDP port (or ICMP code)
id.resp_h	addr	Responding endpoint's IP address (AKA Resp)
id.resp_p	port	Responding endpoint's TCP/UDP port (or ICMP code)
proto	proto	Transport layer protocol of connection
service	string	Detected application protocol, if any
duration	interval	Connection length
orig_bytes	count	Orig payload bytes; from sequence numbers if TCP
resp_bytes	count	Resp payload bytes; from sequence numbers if TCP
conn_state	string	Connection state (see conn.log: conn_state table)
local_orig	bool	Is Orig in Site::local_nets? Unset if local_nets is empty.
local_resp	bool	Is Resp in Site::local_nets? Unset if local_nets is empty.
missed_bytes	count	Number of bytes missing due to content gaps
history	string	Connection state history (see conn.log: history table)
orig_pkts	count	Number of Orig packets
orig_ip_bytes	count	Number of Orig IP bytes (via IP total_length header field)
resp_pkts	count	Number of Resp packets
resp_ip_bytes	count	Number of Resp IP bytes (via IP total_length header field)
tunnel_parents	set	If tunneled, connection UID of encapsulating parent(s)

dns.log DNS query/response details

-110 900	3,	soponed details
Field	Туре	Description
ts	time	Timestamp of the DNS request
uid & id		Underlying connection info - See conn.log
proto	proto	Protocol of DNS transaction – TCP or UDP
trans_id	count	16 bit identifier assigned by DNS client; responses match
query	string	Domain name subject of the query
qclass	count	Value specifying the query class
qclass_name	string	Descriptive name of the query class (e.g. C_INTERNET)
qtype	count	Value specifying the query type
qtype_name	string	Descriptive name of the query type (e.g. A, AAAA, PTR)
rcode	count	Response code value in the DNS response
rcode_name	string	Descriptive name of the response code (e.g. NXDOMAIN, NODATA)
AA	bool	Authoritateive Answer. T = server is authoritative for the query
TC	bool	Truncation. T = the message was truncated
RD	bool	Recursion Desired. T = recursive lookup of query requested
RA	bool	Recursion Available. T = server supports recursive queries
Z	count	Reserved field, should be zero in all queries & responses
answers	vector	List of resource descriptions in answer to the query
TTLs	vector	Caching intervals of the answers
rejected	bool	Whether the DNS query was rejected by the server
auth ¹	set	Authoritative responses for the query
addl ¹	set	Additional responses for the query

[1] – If policy/protocols/dns/auth-addl.bro is loaded

We at Broala are committed to helping you understand Bro to the fullest so you can be a monitoring hero. The latest version of this guide is always freely availlable at: https://github.com/broala/bro-cheatsheets

conn.log: conn_state

	mogr com_state
State	e Meaning
S0	Connection attempt seen, no reply
S1	Connection established, not terminated (0 byte counts)
SF	Normal establish & termination (>0 byte counts)
REJ	Connection attempt rejected
S2	Established, Orig attempts close, no reply from RESP.
S3	Established, Resp attempts close, no reply from ORIG.
RSTO	Established, Orig aborted (RST)
RSTR	Established, Resp aborted (RST)
RSTO	SO Orig sent SYN then RST; no Resp SYN-ACK
RSTRI	H Resp sent SYN-ACK then RST; no Orig SYN
SH	Orig sent SYN then FIN; no Resp SYN-ACK ("half-open")
SHR	Resp sent SYN-ACK then FIN; no Orig SYN
ОТН	No SYN, not closed. Midstream traffic. Partial connection.

conn.log: history Orig UPPERCASE, Resp lowercase, uniq-ed

Letter	Meaning
S	a SYN without the ACK bit set
н	a SYN-ACK ("handshake")
Α	a pure A CK
D	packet with payload ("data")
F	packet with FIN bit set
R	packet with R ST bit set
С	packet with a bad checksum
I	Inconsistent packet (Both SYN & RST)

capture_loss.log Estimate of packet loss

Field	Туре	Description
ts	time	Timestamp of the end of the measurement
ts_delta	interval	Time difference from previous measurement
peer	string	Name of the Bro instance reporting loss
gaps	count	ACKs seen without seeing the data being ACKed
acks	count	Total number of TCP ACKs
percent_loss	double	Estimate of loss: gaps/acks

dhcp.log DHCP lease activity

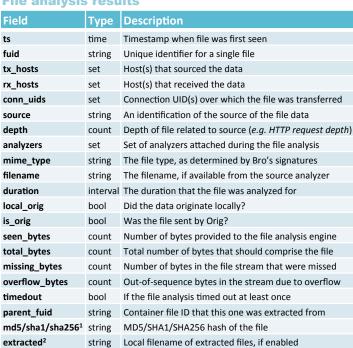
Field	Туре	Description
ts	time	Timestamp of the DHCP lease request
uid & id		Underlying connection info - See conn.log
mac	string	Client's hardware address
assigned_ip	addr	Client's actual assigned IP address
lease_time	interval	IP address lease time
trans_id	count	Identifier assigned by the client; responses match

dnp3.log Distributed Network Protocol (ICS)

Field	Туре	Description
ts	time	Timestamp of the DNP3 request
uid & id		Underlying connection info - See conn.log
fc_request	string	The name of the request function message
fc_reply	string	The name of the reply function message
iin	count	Response's "internal indication number"

files.log

File analysis results



[1] - If base/files/hash/main.bro is loaded

[2] – If base/files/extract/main.bro is loaded

ftp.log FTP request/reply details

Field	Туре	Description
ts	time	Timestamp of the FTP command
uid & id		Underlying connection info - See conn.log
user	string	Username for the FTP session
password	string	Password for the FTP session
command	string	Command issued by the client
arg	string	Any command arguments
mime_type	string	File type if there's a file transfer
file_size	count	Size of transferred file
reply_code	count	Reply code from server in response to the command
reply_msg	string	Reply message from server in response to the command
data_channel	record	Information about the data channel (orig, resp, is passive)
fuid¹	string	File unique ID

[1] – If base/protocols/ftp/files.bro is loaded

irc.log IRC communication details

ind communication details		
Field	Туре	Description
ts	time	Timestamp of the IRC command
uid & id		Underlying connection info - See conn.log
nick	string	Nickname given for this connection
user	string	Username given for this connection
command	string	Command given by the client
value	string	Value for the command given by the client
addl	string	Any additional data for the command
fuid ¹	string	File unique ID

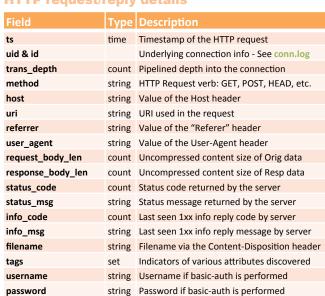
[1] - If base/protocols/irc/files/bro is loaded

Note: base/protocols/irc/dcc-send.bro adds several DCC-related fields



http.log

HTTP request/reply details



vector File unique IDs from Orig

vector File unique IDs from Resp

vector File types from Orig

vector File types from Resp

server_header_names2 vector The names of HTTP headers sent by Resp

Headers indicative of a proxied request

vector The names of HTTP headers sent by Orig

vector Variable names extracted from cookies

vector Variable names extracted from the URI

www.broala.con

[1] - If base/protocols/http/entities.bro is loaded

[2] – If policy/protocols/http/header-names.bro is loaded

 $\hbox{[3]-If policy/protocols/http/var-extraction-uri.bro is loaded}$

intel.log

proxied

orig_fuids1

resp_fuids1

cookie vars³

uri vars³

orig mime types1

resp_mime_types1

client_header_names2

Hits on indicators from the intel framework

Field	Туре	Description
ts	time	Timestamp of the intelligence hit
uid & id		Underlying connection info - See conn.log
fuid	string	The UID for a file associated with this hit, if any
file_mime_type	string	A mime type if the hit is related to a file
file_desc	string	Additional context for file, if available
seen.indicator	string	The intelligence indicator
seen.indicator_type	string	The type of data the indicator represents
seen.host	addr	If the indicator was Intel::ADDR, the address
seen.where	string	Where the data was discovered
seen.node	string	The name of the node that discovered the match
sources	set	Sources which supplied data for this match

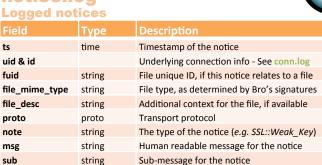
modbus.log PLC requests (ICS)

Field	Туре	Description
ts	time	Timestamp of the PLC request
uid & id		Underlying connection info - See conn.log
func	string	Function message that was sent
exception	string	Exception if there was a failure

notice.log **Logged notices**

ts

dst



Source address

Destination address

Associated port, if any

count Associated count or status code peer_descr string Name of the node that raised this notice actions Actions applied to this notice suppress_for interval Length of time dupes should be suppressed dropped1 If the src IP was blocked

remote_location² geo_location GeoIP data about the hosts involved [1] - If base/frameworks/notice/actions/drop.bro is loaded

[2] – If base/frameworks/notice/actions/add-geodata.bro is loaded

radius.log **RADIUS** authentication attempts

addr

addr

port

Field	Type	Description
ts	time	Timestamp of the authentication attempt
uid & id		Underlying connection info - See conn.log
username	string	The username of the user attempting to authenticate
mac	string	The MAC address of the client (e.g. for wireless)
remote_ip	addr	The IP address of the client (e.g. for VPN)
connect_info	string	Additional connect information, if available
result	string	Whether the attempt succeeded or failed

smtp.log **SMTP** transactions

Field	Туре	Description
ts	time	Timestamp when the message was first seen
uid & id		Underlying connection info - See conn.log
trans_depth	count	Transaction depth if there are multiple msgs
helo	string	Contents of the HELO header
mailfrom	string	Contents of the MAIL FROM header
rcptto	set	Contents of the RCPT TO header
date	string	Contents of the DATE header
from	string	Contents of the FROM header
to	set	Contents of the TO header
reply_to	string	Contents of the ReplyTo header
msg_id	string	Contents of the MsgID header
in_reply_to	string	Contents of the In-Reply-To header
subject	string	Contents of the Subject header
x_originating_ip	addr	Contents of the X-Originating-IP header
first_received	string	Contents of the first Received header
second_received	string	Contents of the second Received header
last_reply	string	Last server to client message
path	vector	Message transmission path, from headers
user_agent	string	Value of the client User-Agent header
tls	bool	Indicates the connection switched to TLS
fuids ¹	vector	File unique IDs seen attached to this message
is_webmail ²	bool	If the message was sent via webmail

[1] - If base/protocols/smtp/files.bro is loaded

[2] – If policy/protocols/smtp/software.bro is loaded





weird.log

Anomalies and protocol violations

Field	Туре	Description
ts	time	Timestamp of message
uid & id		Underlying connection info - See conn.log
name	string	The name of the weird that occurred
addl	string	Additional information accompanying the weird, if any
notice	bool	Indicate if this weird was also turned into a notice
peer	string	The peer that generated this weird

snmp.log SNMP messages

Field	Туре	Description
ts	time	Timestamp when the message was first seen
uid & id		Underlying connection info - See conn.log
duration	interval	Time between the first and last seen packet
version	string	SNMP version (v1, v2c, v3)
community	string	The community string of the first SNMP packet
get_requests	count	Number of GetRequest/GetNextRequest packets
get_bulk_requests	count	Number of GetBulkRequest packets
get_responses	count	Number of GetResponse/Response packets
set_requests	count	Number of SetRequest packets
display_string	string	A system description of Resp
up_since	time	Timestamp that Resp has been up since

socks.log **SOCKS** proxy requests

Field	Туре	Description	
ts	time	Timestamp of the SOCKS proxy request	
uid & id		Underlying connection info - See conn.log	
version	count	SOCKS protocol version	
user	string	Username for proxy auth, if available	
password	string	Password for proxy auth, if available	
status	string	Server status for the proxy request	
request.host	addr	Client requested address	
request.name	string	Client requested name	
request_p	port	Client requested port	
bound.host	addr	Server bound address	
bound.name	string	Server bound name	
bound_p	port	Server bound port	

software.log

Software identified by the software framework

Field	Туре	Description
ts	time	Timestamp of the first software detection
host	addr	IP address running the software
host_p	port	Port on which the software is running (for servers)
software_type	Software::Type	Type of software (e.g. HTTP::SERVER)
name	string	Name of the software
version	Software::Version	Version of the software
unparsed_version	string	The full, unparsed version of the software
url ¹	string	Root URL where the software was found

[1] - If policy/protocols/http/detect-webapps.bro is loaded

ssh.log SSH handshakes

Field	Туре	Description	
ts	time	Timestamp when the SSH conn was detected	
uid & id		Underlying connection info - See conn.log	
version	count	SSH major version (1 or 2)	
auth_success	bool	Did the auth succeed? Unset if undetermined	
direction	Direction	Inbound or outbound connection	
client	string	Software string from the client	
server	string	Software string from the server	
cipher_alg	string	The negotiated encryption algorithm	
mac_alg	string	The negotiated MAC (signing) algorithm	
compression_alg	string	The negotiated compression algorithm	
kex_alg	string	The negotiated key exchange algorithm	
host_key_alg	string	The server's host key algorithm	
host_key	string	The server's host key fingerprint	
remote_location1	geo_location	GeoIP data for the "remote" endpoint	

^{[1] –} If policy/protocols/ssh/geo-data.bro is loaded

ssl.log SSL handshakes

Field	Туре	Description
ts	time	Timestamp when the SSL connection was detected
uid & id		Underlying connection info - See conn.log
version	string	SSL version that the server offered
cipher	string	SSL cipher suite that the server chose
curve	string	Elliptic curve the server chose if using ECDH/ECDHE
server_name	string	Value of the Server Name Indicator SSL extension
session_id	string	Session ID offered by client for session resumption
resumed	bool	Flag that indicates the session was resumed
last_alert	string	Last alert that was seen during the connection
next_protocol	string	Next protocol the server chose using the application layer next protocol extension, if seen.
established	bool	Was this connection established successfully?
cert_chain1	vector	Chain of certificates offered by the server
cert_chain_fuids1	vector	File UIDs for certs in cert_chain.
client_cert_chain1	vector	Chain of certificates offered by the client
client_cert_chain_ fuids ¹	vector	File UIDs for certs in client_cert_chain .
subject1	string	Subject of the X.509 cert offered by the server
issuer ¹	string	Subject of the signer of the server cert
client_subject1	string	Subject of the X.509 cert offered by the client
client_issuer1	string	Subject of the signer of the client cert
validation_status ²	string	Certificate validation result for this handshake
ocsp_status ²	string	OCSP validation result for this handshake
ocsp_response ²	string	OCSP response as a string
notary ³	CertNotary:: Response	A response from the ICSI certificate notary.

- [1] If base/protocols/ssl/files.bro is loaded
- [2] If policy/protocols/ssl/validate-certs.bro is loaded
- [3] If policy/protocols/ssl/notary.bro is loaded

tunnel.log

Details of encapsulating tunnels

Details 0	Cilco	ipadiating turniers
Field	Type	Description
ts	time	Timestamp tunnel was detected
uid & id		Underlying connection info - See conn.log
tunnel_type	string	The type of tunnel (e.g. Teredo, IP)
action	string	The activity that occurred (discovered, closed)



reporter.log Bro internal errors and warnings

Field	Туре	Description
ts	time	Message timestamp, if available (0 otherwise)
level	string	Message severity (info, warning, error, etc.)
message	string	Message text
location	string	The script location of the event, if available

x509.log SSL certificate details

Field	Туре	Description
ts	time	Time when the cert was seen
id	string	File unique ID
certificate.version	count	Cert version number
certificate.serial	string	Cert serial number
certificate.subject	string	Cert subject
certificate.issuer	string	Cert issuer
certificate.not_valid_before	time	Time the cert is valid from
certificate.not_valid_after	time	Time the cert is valid until
certificate.key_alg	string	Name of the key algorithm
certificate.sig_alg	string	Name of the signature algorithm
certificate.key_type	string	Key type (RSA, DSA or EC)
certificate.key_length	count	Key length, in bits
certificate.exponent	string	Exponent, if RSA
certificate.curve	string	Curve, if EC
san.dns	string_vec	List of DNS entries in Subject Alternative Name (SAN)
san.uri	string_vec	List of URI entries in SAN
san.email	string_vec	List of email entries in SAN
san.ip	addr_vec	List of IP entries in SAN
basic_constraints.ca	bool	CA flag set?
basic_constraints.path_len	count	Maximum path length

Other Logs

Log	Description
cluster	Diagnostics for cluster operation
communication	Diagnostics for Bro inter-process communications
dpd	Diagnostics for dynamic protocol detection
known_certs	Observed local SSL certs. Each is logged once/day
known_devices	Observed local devices. Logged once/day
known_hosts	Observed local active IPs. Logged once/day
known_services	Observed local listening services. Logged once/day
loaded_scripts	A list of Bro scripts that were loaded at startup
mysql	MySQL requests and responses
packet_filter	Any filters to limit the traffic being analyzed
prof	Performance profiling data
signatures	Hits from the Signatures framework
stats	Diagnostics such as mem usage, packets seen, etc.
syslog	Syslog messages
traceroute	Hosts running traceroute

In order to promote its wide distribution, this work is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (http://creativecommons.org/licenses/by-nc-sa/4.0/).