# Bro 2.3 Logs

## conn.log
### IP, TCP, UDP and ICMP connection details

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| uid | string | Unique ID of 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 | Dynamically detected application protocol, if any |
| duration | interval | Connection length |
| orig_bytes | count | Originator payload bytes; from sequence numbers if TCP |
| resp_bytes | count | Responder payload bytes; from sequence numbers if TCP |
| conn_state | string | Connection state (see conn.log: conn_state table) |
| local_orig | bool | If conn originated locally T; if remotely F. If Site::local_nets empty, always unset. |
| missed_bytes | count | Number of missing bytes in 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) |
| orig_cc | string | ORIG GeoIP Country Code |
| resp_cc | string | RESP GeoIP Country Code |

## dns.log
### DNS query/response details

| Field | Type | 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 | 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. NOERROR, NXDOMAIN) |
| QR | bool | Was this a query (T) or a response (F)? |
| AA | bool | T: server is authoritative for query |
| TC | bool | T: message was truncated |
| RD | bool | Recursion Desired. T = request recursive lookup of query |
| 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 |

## conn.log: conn_state

| State | 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) |
| RSTOS0 | ORIG sent SYN then RST; no RESP SYN-ACK |
| RSTRH | 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 |
| OTH | 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 |
| H | a SYN-ACK ("handshake") |
| A | a pure ACK |
| D | packet with payload ("data") |
| F | packet with FIN bit set |
| R | packet with RST bit set |
| C | packet with a bad checksum |
| I | Inconsistent packet (Both SYN & RST) |

## capture_loss.log
### Estimate of packet loss

| Field | Type | Description |
|---|---|---|
| ts | time | Measurement timestamp |
| ts_delta | interval | Time difference from previous measurement |
| peer | string | Name of the Bro instance reporting loss |
| gaps | count | ACKs seen without seeing data being ACKed |
| acks | count | Total number of TCP ACKs |
| percent_loss | string | gaps/acks, as a percentage. Estimate of loss. |

## dhcp.log
### DHCP lease activity

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of 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 |

44

© Broala LLC.

Version: 2.0

# Bro 2.3 Logs

## dnp3.log
### Distributed Network Protocol (industrial control)

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| 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

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when file was first seen |
| fuid | string | Unique identifier for a single file |
| tx_hosts | set | if transferred via network, host(s) that sourced the data |
| rx_hosts | set | if transferred via network, host(s) that received the data |
| conn_uids | set | Connection UID(s) over which the file was transferred |
| source | string | An identification of the source of the file data |
| depth | count | Depth of file related to source; eg: SMTP MIME attachment depth; HTTP depth of the request |
| analyzers | set | Set of analysis types done during file analysis |
| mime_type | string | The file type, as determined by Bro's signatures |
| filename | string | If available, filename from source; frequently the "Content-Disposition" headers in network protocols |
| duration | interval | The duration the file was analyzed for |
| local_orig | bool | If transferred via network, did data originate locally? |
| is_orig | bool | If transferred via network, was file sent by the originator? |
| seen_bytes | count | Number of bytes provided to file analysis engine |
| total_bytes | count | Total number of bytes that should comprise the file |
| missing_bytes | count | Number of bytes in the file stream missed; eg: dropped packets |
| overflow_bytes | count | Number of not all-in-sequence bytes in the file stream delivered to file analyzers due to reassembly buffer overflow |
| timedout | bool | If the file analysis time out at least once per file |
| parent_fuid | string | ID associated with a container file from which this one was extracted as a part of the analysis |
| md5/sha1/sha256 | string | MD5/SHA1/SHA256 hash of file, if enabled |
| extracted | string | Local filename of extracted files, if enabled |

## ftp.log
### FTP request/reply details

| Field | Type | Description |
|---|---|---|
| ts | time | Command timestamp |
| uid & id | | Underlying connection info - See conn.log |
| user | string | Username for current FTP session |
| password | string | Password for current FTP session |
| command | string | Command issued by the client |
| arg | string | Command argument if present |
| mime_type | string | Libmagic sniffed 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) |
| fuid | string | File unique ID |

## http.log
### HTTP request/reply details

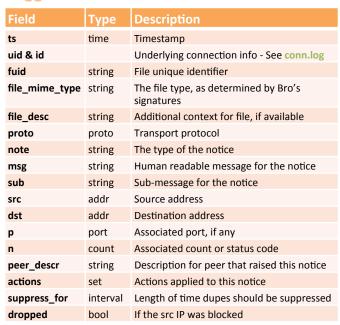| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of request |
| uid & id | | Underlying connection info - See conn.log |
| trans_depth | count | Pipelined depth into the connection |
| method | string | HTTP Request verb: GET, POST, HEAD, etc. |
| host | string | Value of the HOST header |
| uri | string | URI used in the request |
| referrer | string | Value of the "referer" header |
| user_agent | string | Value of the User-Agent header |
| request_body_len | count | Actual uncompressed content size of the data transferred from the **client** |
| response_body_len | count | Actual uncompressed content size of the data transferred from the **server** |
| status_code | count | Status code returned by the server |
| status_msg | string | Status message returned by the server |
| info_code | count | Last seen 1xx info reply code by server |
| info_msg | string | Last seen 1xx info reply message by server |
| filename | string | Via the Content-Disposition server header |
| tags | set | Indicators of various attributes discovered |
| username | string | If basic-auth is performed for the request |
| password | string | If basic-auth is performed for the request |
| proxied | set | Headers that might indicate a proxied request |
| orig_fuids | vector | An ordered vector of file unique IDs from orig |
| orig_mime_types | vector | An ordered vector of mime types from orig |
| resp_fuids | vector | An ordered vector of file unique IDs from resp |
| resp_mime_types | vector | An ordered vector of mime types from resp |

## intel.log
### Hits on indicators from the intel framework

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of 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.where | string | Where the data was discovered |
| sources | set | Sources which supplied data for this match |

## irc.log
### IRC communication details

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| 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 |
| dcc_file_name | string | DCC filename requested |
| dcc_file_size | count | Size of the DCC transfer as indicated by the sender |
| dcc_mime_type | string | Sniffed mime type of the file |
| fuid | string | File unique ID |

45

# Bro 2.3 Logs

## notice.log
### Logged notices

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| uid & id | | Underlying connection info - See conn.log |
| fuid | string | File unique identifier |
| file_mime_type | string | The file type, as determined by Bro's signatures |
| file_desc | string | Additional context for file, if available |
| proto | proto | Transport protocol |
| note | string | The type of the notice |
| msg | string | Human readable message for the notice |
| sub | string | Sub-message for the notice |
| src | addr | Source address |
| dst | addr | Destination address |
| p | port | Associated port, if any |
| n | count | Associated count or status code |
| peer_descr | string | Description for peer that raised this notice |
| actions | set | Actions applied to this notice |
| suppress_for | interval | Length of time dupes should be suppressed |
| dropped | bool | If the src IP was blocked |

## radius.log
### RADIUS authentication attempts

| 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 auth |
| 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 | Type | 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 |
| fuids | vector | File unique IDs seen attached to this msg |
| is_webmail | bool | If the message was sent via webmail |

## modbus.log
### PLC requests (industrial control)

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

## snmp.log
### SNMP messages

| Field | Type | 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 the responder |
| up_since | time | Timestamp the responder has been up since |

## socks.log
### SOCKS proxy requests

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of request |
| uid & id | | Underlying connection info - See conn.log |
| version | count | Protocol version of SOCKS |
| user | string | Username for the proxy, if available |
| status | string | Server status for the attempt using proxy |
| 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 | Type | Description |
|---|---|---|
| ts | time | Timestamp of the detection |
| host | addr | IP address running the software |
| host_p | port | Port on which the software is running (for servers) |
| software_type | string | Type of software (e.g. HTTP::SERVER) |
| name | string | Name of the software |
| version.major | count | Major version number of the software |
| version.minor | count | Minor version number of the software |
| version.minor2 | count | Minor subversion number of the software |
| version.minor3 | count | Minor update number of the software |
| version.addl | string | Additional version string (e.g. beta42) |
| unparsed_version | string | The full, unparsed version of the software |

46

# Bro 2.3 Logs

## ssh.log
### SSH handshakes

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when the SSH connection was detected |
| uid & id | | Underlying connection info - See conn.log |
| status | string | If the login was heuristically guessed to be "success" or "failure". |
| direction | string | Outbound or inbound connection |
| client | string | Software string from the client |
| server | string | Software string from the server |
| resp_size | count | Amount of data returned by the server |

## ssl.log
### SSL handshakes

| Field | Type | 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 |
| last_alert | string | Last alert that was seen during the connection |
| established | bool | Was this connection established successfully? |
| cert_chain | vector | Chain of certificates offered by the server |
| cert_chain_fuids | vector | File unique IDs for certs in cert_chain. See files.log |
| client_cert_chain | vector | Chain of certificates offered by the client |
| client_cert_chain_fuids | vector | File UIDs for certs in client_cert_chain. See files.log |
| subject | string | Subject of the X.509 cert offered by the server |
| issuer | string | Subject of the signer of the server cert |
| client_subject | string | Subject of the X.509 cert offered by the client |
| client_issuer_subject | string | Subject of the signer of the client cert |
| validation_status | string | Certificate validation result for this handshake |
| ocsp_status | string | Result of OCSP validation for this handshake |
| ocsp_response | string | OCSP response as a string |

## tunnel.log
### Details of encapsulating tunnels

| 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) |

## weird.log
### Anomalies and protocol violations

| Field | Type | 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 |

## reporter.log
### Bro internal errors and warnings

| Field | Type | 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 where the event occurred, if available |

## x509.log
### SSL certificate details

| Field | Type | Description |
|---|---|---|
| ts | time | Time when the cert was seen |
| id | string | File unique ID. See files.log |
| certificate.version | count | Version number |
| certificate.serial | string | Serial number |
| certificate.issuer | string | Issuer |
| certificate.not_valid_before | time | Time before when the cert is invalid |
| certificate.not_valid_after | time | Time after when the cert is invalid |
| certificate.key_alg | string | Name of the key algorithm |
| certificate.sig_alg | string | Name of the signature algorithm |
| certificate.key_type | string | Key type (either 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 |
|---|---|
| app_stats | Statistics on usage of popular web apps |
| cluster | Diagnostics for cluster operation |
| communication | Diagnostics for 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. Each is logged once/day |
| known_hosts | Observed local active IPs. Each is logged once/day |
| known_services | Observed local services. Each is logged once/day |
| loaded_scripts | A list of scripts that were loaded at startup |
| packet_filter | Any filters to limit the traffic being analyzed |
| stats | Diagnostics such as mem usage, packets seen, etc. |
| syslog | Syslog messages |
| traceroute | Hosts running traceroute |

Version: 2.0

# Bro Logs

## app_stats.log
### Statistics on usage of popular web apps

| Field | Type | Description |
|---|---|---|
| ts | time | Measurement timestamp |
| ts_delta | interval | Time difference from previous measurement |
| app | string | Name of application (YouTube, Netflix, etc.) |
| uniq_hosts | count | Number of unique hosts that used app |
| hits | count | Number of visits to app |
| bytes | count | Total bytes transferred to/from app |

## capture_loss.log
### Estimate of packet loss

| Field | Type | Description |
|---|---|---|
| ts | time | Measurement timestamp |
| ts_delta | interval | Time difference from previous measurement |
| peer | string | Name of the Bro instance reporting loss |
| gaps | count | ACKs seen without seeing data being ACKed |
| acks | count | Total number of TCP ACKs |
| percent_loss | string | gaps/acks, as a percentage. Estimate of loss. |

## dhcp.log
### DHCP lease activity

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of request |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. 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 |

## conn.log
### IP, TCP, UDP and ICMP connection details

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| uid | string | Unique ID of 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 | transport_proto | Transport layer protocol of connection |
| service | string | Dynamically detected application protocol, if any |
| duration | interval | Time of last packet seen – time of first packet seen |
| orig_bytes | count | Originator payload bytes; from sequence numbers if TCP |
| resp_bytes | count | Responder payload bytes; from sequence numbers if TCP |
| conn_state | string | Connection state (see conn.log:conn_state table) |
| local_orig | bool | If conn originated locally T; if remotely F. If Site::local_nets empty, always unset. |
| missed_bytes | count | Number of missing bytes in 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) |
| orig_cc | string | ORIG GeoIP Country Code |
| resp_cc | string | RESP GeoIP Country Code |

## dns.log
### DNS query/response details

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of the DNS request |
| uid | string | Unique id of the connection |
| id | record | ID record with orig/resp host/port. 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 | 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. NOERROR, NXDOMAIN) |
| QR | bool | Was this a query or a response? T = response, F = query |
| AA | bool | Authoritative Answer. T = server is authoritative for query |
| TC | bool | Truncation. T = message was truncated |
| RD | bool | Recursion Desired. T = request recursive lookup of query |
| 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 |

## conn.log: conn_state

| State | 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) |
| RSTOS0 | ORIG sent SYN then RST; no RESP SYN-ACK |
| RSTRH | 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 |
| OTH | 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 |
| H | a SYN-ACK ("handshake") |
| A | a pure ACK |
| D | packet with payload ("data") |
| F | packet with FIN bit set |
| R | packet with RST bit set |
| C | packet with a bad checksum |
| I | Inconsistent packet (Both SYN & RST) |

48

# Bro Logs

## known_certs.log
### Observed local Certs; logged 1xDay

| Field | Type | Description |
|---|---|---|
| ts | time | Measurement timestamp |
| host | addr | Address that offered the certificate |
| port_num | port | If server, port that server listening on |
| subject | string | Certificate subject |
| issuer_subject | string | Certificate issuer subject |
| serial | string | Serial number for the certificate |

## known_services.log
### Observed local services; logged 1xDay

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| host | addr | Host address on which the service is running |
| port_num | port | Port number on which the service is running |
| port_proto | transport_proto | Transport-layer protocol service uses |
| service | set | Set of protocol(s) that match the service's connection payloads |

## modbus.log
### PLC requests (industrial control)

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of request |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| func | string | Function message that was sent |
| exception | string | Exception if there was a failure |

## notice.log
### Logged notices

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| fuid | string | File unique identifier |
| file_mime_type | string | Libmagic sniffed file type |
| file_desc | string | Additional context for file, if available |
| proto | transport_proto | Transport protocol |
| note | string | The type of the notice |
| msg | string | Human readable message for the notice |
| sub | string | Sub-message for the notice |
| src | addr | Source address |
| dst | addr | Destination address |
| p | port | Associated port, if any |
| n | count | Associated count or status code |
| peer_descr | string | Description for peer that raised this notice |
| actions | set | Actions applied to this notice |
| suppress_for | interval | Length of time dupes should be suppressed |
| dropped | bool | If the **src** IP was blocked |

## known_hosts.log
### Observed local active IPs; logged 1xDay

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp first seen |
| host | addr | IP Address of host |

## radius.log
### Radius authentication details

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of the detection |
| uid | string | Unique ID for the connection |
| id | conn_id | ID record with orig/resp host/port. See conn.log |
| username | string | The username, if present |
| mac | string | MAC address, if present |
| remote_ip | addr | Remtoe IP address, if present |
| connect_info | string | Connect info, if present |
| result | string | Successful or failed authentication |
| logged | bool | Whether this has already been logged & ignored |

## reporter.log
### Bro internal errors and warnings

| Field | Type | Description |
|---|---|---|
| ts | time | Message timestamp |
| level | string | Message severity (Info, warning, error, etc.) |
| message | string | Message text |
| location | string | The script location where tevent occurred, if available |

## smtp.log
### SMTP transactions

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when the message was first seen |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| trans_depth | count | Depth of message transaction if multiple messages transferred |
| 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 message that the server sent to the client |
| path | vector | Message transmission path, extracted from the headers |
| user_agent | string | Value of the User-Agent header from the client |
| tls | bool | Connection has switched to using TLS |
| fuids | vector | File unique IDs seen attached to this message |
| is_webmail | bool | Indicates if the message was sent through a webmail interface |

49

# Bro Logs

## signatures.log
### Matches from the signature framework

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of match |
| src_addr | addr | Host triggering the signature match event |
| src_port | port | Host port on which the match occurred |
| dst_addr | addr | Host which was sent the matching payload |
| dst_port | port | Port which was sent the matching payload |
| note | string | Notice associated with the signature event |
| sig_id | string | Name of the signature that matched |
| event_msg | string | More descriptive message of the event |
| sub_msg | string | Extracted payload data or extra message |
| sig_count | count | Number of sigs |
| host_count | count | Number of hosts |

## snmp.log
### SNMP communication

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp tunnel was detected |
| uid | string | Connection unique id |
| id | conn_id | ID record with orig/resp host/port. See conn.log |
| duration | interval | Amount of time between first/latest packet in session |
| version | string | The version of SNMP being used |
| community | string | Community string of the first SNMP packet associated w/ session; v1 & v2c only |
| get_requests | count | Number of variable bindings in GetRequest/Next |
| get_bulk_requests | count | Number of variable bindings in GetBulkRequest PDU |
| get_responses | count | Number of variable bindings in GetResponse/Response PDUs |
| set_requests | count | Number of variable bindings in SetRequest PDUs |
| display_string | string | System description of the SNMP responder endpoint |
| up_since | time | Time the SNMP responder claims it has been up since |

## ssl.log
### SSL handshakes (v2.2 only; v2.3 x509.log)

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when the SSL connection was detected |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| version | string | SSL version that the server offered |
| cipher | string | SSL cipher suite that the server chose |
| server_name | string | Value of the Server Name Indicator SSL extension |
| session_id | string | Session ID offered by the client for session resumption |
| subject | string | Subject of the X.509 cert offered by the server |
| issuer_subject | string | Signer Subject of the cert offered by the server |
| not_valid_before | time | NotValidBefore field value from the server cert |
| not_valid_after | time | NotValidAfter field value from the server cert |
| last_alert | string | Last alert that was seen during the connection |
| client_subject | string | Subject of the X.509 cert offered by the client |
| clnt_issuer_subject | string | Subject of the signer of the cert offered by the client |
| cert_hash | string | MD5 hash of the raw server certificate |
| validation_status | vector | Certificate validation for this connection |

## stderr.log / stdout.log

| Description |
|---|
| **Error / output logging** - LogAscii::output_to_stdout = F &redef |

## software.log
### Software identified by the software framework

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of the detection |
| host | addr | IP address running the software |
| host_p | port | Port on which the software is running (for servers) |
| software_type | string | Type of software (e.g. HTTP::SERVER) |
| name | string | Name of the software |
| version.major | count | Major version number of the software |
| version.minor | count | Minor version number of the software |
| version.minor2 | count | Minor subversion number of the software |
| version.minor3 | count | Minor update number of the software |
| version.addl | string | Additional version string (e.g. beta42) |
| unparsed_version | string | The full, unparsed version of the software |

## ssh.log
### SSH handshakes

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when the SSH connection was detected |
| uid | string | Connection unique ID |
| id | record | ID record with orig/resp host/port. See conn.log |
| status | string | If the login was heuristically guessed to be a "success" or a "failure". |
| direction | string | Outbound or inbound connection |
| client | string | Software string from the client |
| server | string | Software string from the server |
| resp_size | count | Amount of data returned by the server |

## socks.log
### SOCKS proxy requests

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp of request |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| version | count | Protocol version of SOCKS |
| user | string | Username for proxy, if available |
| status | string | Server status for the attempt using proxy |
| 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 |

## syslog.log
### Syslog messages

| Field | Type | Description |
|---|---|---|
| ts | time | Timestamp when the message was seen |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| proto | transport_proto | Protocol over which message was seen. Only UDP is currently supported. |
| facility | string | Syslog facility for the message |
| severity | string | Syslog severity for the message |
| message | string | The plain text syslog message |

# Bro Logs

## traceroute.log
### Hosts running traceroute

| Field | Type | Description |
|-------|------|-------------|
| ts | time | Timestamp traceroute was detected |
| src | addr | Address initiating the traceroute |
| dst | addr | Destination address of the traceroute |
| proto | string | Protocol used for the traceroute |

## tunnel.log
### Details of encapsulating tunnels

| Field | Type | Description |
|-------|------|-------------|
| ts | time | Timestamp tunnel was detected |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. See conn.log |
| tunnel_type | string | The type of tunnel (e.g. Teredo, IP) |
| action | string | The activity that occurred (discovered, closed) |

## x509.log
### x509 Certificate Analyzer Output

| Field | Type | Description |
|-------|------|-------------|
| ts | time | Timestamp of the detection |
| id | String | File id of this certificate |
| certificate . | record | Certificate details |
| .version | count | Version number |
| .serial | string | Serial number |
| .issuer | string | Certificate issuer |
| .not_valid_before | time | Timestamp before when certificate is not valid |
| .not_valid_after | time | Timestamp after when certificate is not valid |
| .key_alg | string | Name of the key algorithm |
| .sig_alg | string | Name of the signature algorithm |
| .key_type | string | Key type, if key parseable openssl (rsa, dsa or ec) |
| .key_length | count | Key length in bits |
| .exponent | string | Exponent, if RSA-certificate |
| .curve | string | Curve, if EC-certificate |
| san. | record | Subject Alternative Name |
| .dns | string_vec | List of DNS entries in the SAN |
| .uri | string_vec | List of URI entries in the SAN |
| .email | string_vec | List of email entries in the SAN |
| .ip | addr_vec | List of IP entries in the SAN |
| .other_fields | bool | True if certificate contained other, unrecognized fields |
| basicconstraints. | record | Basic constraints extension of the certificate |
| .ca | bool | CA fla set? |
| .path_len | count | Maximum path length |
| logcert | bool | T (present if policy/protocols/ssl/log-hostcerts-only.bro) |

## weird.log
### Anomalies and protocol violations

| Field | Type | Description |
|-------|------|-------------|
| ts | time | Timestamp of message |
| uid | string | Connection unique id |
| id | record | ID record with orig/resp host/port. 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 |

# Contact Critical Stack

| Command | Description |
|---------|-------------|
| Phone: | 202-559-5200 |
| Email: | info@CriticalStack.com |
| Web: | http://www.CriticalStack.com |
| Git: | https://github.com/CriticalStack/ |
| Twitter: | @CriticalStack |
| pgp | 0xc255d63501b80df9 |

# Index

5 of 1

Version: 2.3