

ZEEK® LOGS

conn.log | IP, TCP, UDP, ICMP connection

| FIELD | TYPE | DESCRIPTION |
|----------------|----------|--|
| ts | time | Timestamp of first packet |
| uid | string | Unique identifier of connection |
| id | record | Connection's 4-tuple of endpoints |
| > id.orig_h | addr | IP address of system initiating connection |
| > id.orig_p | port | Port from which the connection is initiated |
| > id.resp_h | addr | IP address of system responding to connection request |
| > id.resp_p | port | Port on which connection response is sent |
| proto | enum | Transport layer protocol of connection |
| service | string | A comma-separated list of confirmed protocols in the connection |
| duration | interval | How long connection lasted |
| orig_bytes | count | Number of originator IP bytes (via IP total_length header field) |
| resp_bytes | count | Number of payload bytes responder sent |
| conn_state | string | Connection state (see conn.log > conn_state) |
| local_orig | bool | Value=T if connection originated locally |
| local_resp | bool | Value=T if connection responded locally |
| missed_bytes | count | Number of bytes missed (packet loss) |
| history | string | Connection state history (see conn.log > history) |
| orig_pkts | count | Number of packets originator sent |
| orig_ip_bytes | count | Number of originator IP bytes (via IP total_length header field) |
| resp_pkts | count | Number of packets responder sent |
| resp_ip_bytes | count | Number of responder IP bytes (via IP total_length header field) |
| tunnel_parents | table | If tunneled, connection UID value of encapsulating parent(s) |
| ip_proto | string | For IP-based connections, this holds the protocol identifier passed in the IP header |
| orig_ip_addr | string | Link-layer address of originator |
| resp_ip_addr | string | Link-layer address of responder |
| vlan | int | Outer VLAN for connection |
| inner_vlan | int | Inner VLAN for connection |

conn_state

A summarized state for each connection

| | |
|--------|--|
| 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) |
| RSTO50 | Multi-flag packet (SYN or FIN or 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. |

history

Orig UPPERCASE, Resp lowercase

| | |
|---|--|
| 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 packets (e.g., SYN & UID) |
| G | Content Gap |
| T | Multi-flag packet (SYN & FIN or SYN + RST) |
| Q | Retransmitted packet |
| W | Packet with zero window advertisement |
| A | Flipped connection |
| X | Connection analysis partial |

dhcp.log | DHCP lease activity

| FIELD | TYPE | DESCRIPTION |
|-----------------|----------|--|
| ts | time | Earliest time DHCP message observed |
| uids | table | Unique identifiers of DHCP connections |
| client_addr | addr | IP address of client |
| server_addr | addr | IP address of server handing out lease |
| client_port | port | Client port at time of server handing out IP |
| server_port | port | Server port at time of server handing out IP |
| mac | string | Client's hardware address |
| host_name | string | Name given by client in Hostname option 12 |
| client_fqdn | string | FQDN given by client in Client FQDN option 81 |
| domain | string | Domain given by server in option 15 |
| requested_addr | addr | IP address requested by client |
| assigned_addr | addr | IP address assigned by server |
| lease_time | interval | IP address lease interval |
| client_message | string | Message with DHCP_DECLINE so client can tell server why address was rejected |
| server_message | string | Message with DHCP_NAK to let client know why request was rejected |
| msg_types | vector | Duration of DHCP message types seen by transaction |
| duration | interval | Duration of DHCP session |
| client_chaddr | string | Hardware address reported by the client |
| msg_opt | vector | Address originated from server in vendor_class |
| client_software | string | Software reported by client in vendor_class |
| server_software | string | Software reported by server in vendor_class |
| circuit_id | string | DHCP relay agents that terminate circuits |
| agent_remote_id | string | Globally unique ID added by relay agents to identify remote host end of circuit |
| subscriber_id | string | Value independent of physical network connection that provides customer DHCP configuration regardless of physical location |

http.log | HTTP request/reply

| FIELD | TYPE | DESCRIPTION |
|------------------|--------|---|
| ts | time | Timestamp for when request happened |
| uid & id | string | Underlying connection info - See conn.log |
| trans_depth | count | Pipelined depth into connection |
| method | string | Verb used in HTTP request (GET, POST, etc) |
| host | string | Value of HOST header |
| uri | string | URI used in request |
| referrer | string | Value of referrer header |
| version | string | Value of version portion of request |
| user_agent | string | Value of User-Agent header from client |
| origin | string | Value of Origin header from client |
| request_body_len | count | Uncompressed data size from client |
| status_code | count | Status code returned by server |
| status_msg | string | Status message returned by server |
| info_code | count | Last seen 'xx info' reply code from server |
| info_msg | string | Last seen 'xx info' reply message from server |
| tags | table | Indicators of various attributes discovered |
| username | string | Username of basic-auth performed for request |
| password | string | Password if basic-auth performed for request |
| proxied | table | All headers indicative of proxied request |
| orig_uids | vector | Ordered vector of file unique IDs |
| orig_filenames | vector | Ordered vector of filenames from client |
| orig_mime_types | vector | Ordered vector of mime types |
| resp_uids | vector | Ordered vector of file unique IDs |
| resp_filenames | vector | Ordered vector of filenames from server |
| resp_mime_types | vector | Ordered vector of mime types |
| client_headers | vector | Vector of HTTP header names sent by client |
| server_headers | vector | Vector of HTTP header names sent by server |
| cookie_vars | vector | Variable names extracted from all cookies |
| uri_vars | vector | Variable names from URI |

irc.log | IRC communication

| FIELD | TYPE | DESCRIPTION |
|---------------|--------|---|
| ts | time | Timestamp when command seen |
| uid & id | string | Underlying connection info - See conn.log |
| nick | string | Nickname given for connection |
| user | string | Username given for connection |
| command | string | Command given by client |
| value | string | Value for command given by client |
| addl | string | Any additional data for command |
| dcc_file_name | string | DCC filename requested |
| dcc_file_size | count | DCC transfer size as indicated by sender |
| dcc_mime_type | string | Sniffed mime type of file |
| uid | string | File unique ID |

kerberos.log | Kerberos authentication

| FIELD | TYPE | DESCRIPTION |
|---------------------|--------|--|
| ts | time | Timestamp for when event happened |
| uid & id | string | Underlying connection info - See conn.log |
| request_type | string | Authentication Service (AS) or Ticket Granting Service (TGS) |
| client | string | Client |
| service | string | Service |
| success | bool | Request result |
| error_msg | string | Error message |
| from | time | Ticket valid from |
| to | time | Ticket valid until |
| cipher | string | Ticket encryption type |
| forwardable | bool | Forwardable ticket requested |
| renewable | bool | Renewable ticket requested |
| client_cert_subject | string | Subject of client certificate, if any |
| client_cert_uid | string | File unique ID of client cert, if any |
| server_cert_subject | string | Subject of server certificate, if any |
| server_cert_uid | string | File unique ID of server cert, if any |
| auth_ticket | string | Ticket hash authorizing request/transaction |
| new_ticket | string | Ticket hash returned by KDC |

ldap.log | LDAP transactions

| FIELD | TYPE | DESCRIPTION |
|----------------|--------|---|
| ts | time | Timestamp for when event happened |
| uid & id | string | Underlying connection info - See conn.log |
| message_id | int | Numeric message ID |
| version | int | LDAP version number |
| opcode | string | Normalized message opcode |
| result | string | Result code |
| diagnostic_msg | string | Result diagnostic |
| object | string | Object identifier |
| argument | string | Message argument |

mysql.log | MySQL

| FIELD | TYPE | DESCRIPTION |
|---------------------|--------|---|
| ts | time | Timestamp for when event happened |
| id | string | File ID of this portable executable file |
| machine | string | Target machine file was compiled for |
| compile_ts | time | Time file was created |
| os | string | Required operating system |
| subsystem | string | Subsystem required to run this file |
| is_exe | bool | Is file an executable, or just an object file? |
| is_64bit | bool | Is file a 64-bit executable? |
| uses_aslr | bool | Does file support Address Space Layout Randomization? |
| uses_dep | bool | Does file support Data Execution Prevention? |
| uses_code_integrity | bool | Does file enforce code integrity checks? |
| uses seh | bool | Does file use structured exception handling? |
| has_import_table | bool | Does file have import table? |
| has_export_table | bool | Does file have export table? |
| has_cert_table | bool | Does file have attribute certificate table? |
| has_debug_data | bool | Does file have debug table? |
| section_names | vector | Names of sections, in order |

pe.log | Portable executable

| FIELD | TYPE | DESCRIPTION |
|---------------------|--------|---|
| ts | time | Timestamp for when event happened |
| id | string | File ID of this portable executable file |
| machine | string | Target machine file was compiled for |
| compile_ts | time | Time file was created |
| os | string | Required operating system |
| subsystem | string | Subsystem required to run this file |
| is_exe | bool | Is file an executable, or just an object file? |
| is_64bit | bool | Is file a 64-bit executable? |
| uses_aslr | bool | Does file support Address Space Layout Randomization? |
| uses_dep | bool | Does file support Data Execution Prevention? |
| uses_code_integrity | bool | Does file enforce code integrity checks? |
| uses seh | bool | Does file use structured exception handling? |
| has_import_table | bool | Does file have import table? |
| has_export_table | bool | Does file have export table? |
| has_cert_table | bool | Does file have attribute certificate table? |
| has_debug_data | bool | Does file have debug table? |
| section_names | vector | Names of sections, in order |

quic.log | QUIC connection updates

| FIELD | TYPE | DESCRIPTION |
|---------------------|--------|---|
| ts | time | Timestamp for when event happened |
| uid & id | string | Underlying connection info - See conn.log |
| version | string | QUIC version found in INITIAL packet |
| client_initial_dcid | string | First Destination Connection ID |
| client_scid | string | Client's Source Connection ID |
| server_scid | string | Server chosen Connection ID |
| server_name | string | Server name extracted from SNI extension |
| client_protocol | string | First protocol extracted from ALPN |
| history | string | Experimental QUIC history |

radius.log | RADIUS authentication attempts

| FIELD | TYPE | DESCRIPTION |
|---------------|----------|---|
| ts | time | Timestamp for when event happened |
| uid & id | string | Underlying connection info - See conn.log |
| username | string | Username, if present |
| mac | string | MAC address, if present |
| framed_addr | addr | Address given to network access server, if present |
| tunnel_client | string | Address (IPv4, IPv6, or FQDN) of initiator end of tunnel, if present |
| connect_info | string | Connect info, if present |
| reply_msg | string | Reply message from server challenge |
| result | string | Successful or failed authentication |
| ttl | interval | Duration between first request and either Access-Accept message or an error |

sip.log | SIP analysis

| FIELD | TYPE | DESCRIPTION |
|-------------------|--------|--|
| ts | time | Timestamp when request happened |
| uid & id | string | Underlying connection info - See conn.log |
| trans_depth | count | Pipelined depth into request/response transaction |
| method | string | Verb used in SIP request (INVITE, etc) |
| uri | string | URI used in request |
| date | string | Contents of Date: header from client |
| request_from | string | Contents of request From: header |
| request_to | string | Contents of To: header |
| response_from | string | Contents of response From: header |
| response_to | string | Contents of response To: header |
| reply_to | string | Contents of Reply-To: header |
| call_id | string | Contents of Call-ID: header from client |
| seq | string | Contents of CSeq: header from client |
| subject | string | Contents of Subject: header from client |
| request_path | vector | Client message transmission path, extracted from headers |
| response_path | vector | Server message transmission path, extracted from headers |
| user_agent | string | Contents of User-Agent: header from client |
| status_code | count | Status code returned by server |
| status_msg | string | Status message returned by server |
| warning | string | Contents of Warning: header |
| request_body_len | count | Contents of Content-Length: header from client |
| response_body_len | count | Contents of Content-Length: header from server |
| content_type | string | Contents of Content-Type: header from server |

* The tags value usually appended to the sender is stripped off and not logged.

smtp.log | SMTP transactions

| FIELD | TYPE | DESCRIPTION |
|------------------|--------|--|
| ts | time | Timestamp when message was first seen |
| uid & id | string | Underlying connection info - See conn.log |
| trans_depth | count | Transaction depth if there are multiple msgs |
| hello | string | Contents of Hello header |
| mailfrom | string | Email addresses found in From header |
| rcptto | table | All addresses found in Rcpt header |
| date | string | Contents of Date header |
| from | string | Contents of From header |
| to | table | Contents of To header |
| cc | table | Contents of Cc header |
| reply_to | string | Contents of Reply-To header |
| msg_id | string | Contents of Message-ID header |
| in_reply_to | string | Contents of In-Reply-To header |
| subject | string | Contents of Subject header |
| x_originating_ip | addr | Contents of X-Originating-IP header |
| x_received | string | Contents of first Received header |
| second_received | string | Contents of second Received header |
| last_reply | string | Last message server sent to client |
| path | vector | Message transmission path, from headers |
| user_agent | string | Value of User-Agent header from client |
| ids | bool | Indicates connection switched to using TLS |
| uids | vector | File unique IDs attached to message |
| is_webmail | bool | If message sent via webmail |

snmp.log | SNMP messages

| FIELD | TYPE | DESCRIPTION |
|-------------------|----------|--|
| ts | time | Timestamp of first packet of SNMP session |
| uid & id | string | Underlying connection info - See conn.log |
| duration | interval | Amount of time between first packet belonging to SNMP session and latest seen |
| version | string | Version of SNMP being used |
| community | string | Community string of first SNMP packet associated with session |
| get_requests | count | Number of variable bindings in GetRequest/GetNextRequest PDUs seen for session |
| get_bulk_requests | count | Number of variable bindings in GetBulkRequest PDUs seen for session |
| get_responses | count | Number of variable bindings in Get-Response/Response PDUs seen for session |
| set_requests | count | Number of variable bindings in SetRequest PDUs seen for session |
| display_string | string | System description of SNMP responder endpoint |
| up_since | time | Time at which SNMP responder endpoint claims it's been up since |

socks.log | SOCKS proxy requests

| FIELD | TYPE | DESCRIPTION |
|-----------|--------|---|
| ts | time | Time when proxy connection detected |
| uid & id | string | Underlying connection info - See conn.log |
| version | count | Protocol version of SOCKS |
| user | string | Username used to request a login to proxy |
| password | string | Password used to request a login to proxy |
| status | string | Server status for attempt at using proxy |
| request | string | Client requested SOCKS address |
| request_p | port | Client requested port |
| bound_p | record | Server bound address |
| bound_p | port | Server bound port |

software.log | Software observed on network

| FIELD | TYPE | DESCRIPTION |
|------------------|--------|---|
| ts | time | Time at which software was detected |
| host | addr | IP address detected running the software |
| host_p | port | Port on which software is running |
| software_type | enum | Type of software detected (e.g., HTTP-SERVER) |
| name | string | Name of software (e.g., Apache) |
| version | record | Software version |
| unparsed_version | string | Full, unparsed version string found |
| url | string | Root URL where software was discovered |

ssh.log | SSH handshakes

| FIELD | TYPE | DESCRIPTION |
|-----------------|--------|---|
| ts | time | Time when SSH connection began |
| uid & id | string | Underlying connection info - See conn.log |
| version | count | SSH major version (1 or 2) |
| auth_success | bool | Authentication result (T=success, F=failure, unset=unknown) |
| auth_attempts | count | Number of authentication attempts observed |
| direction | enum | Direction of connection |
| client | string | Client's version string |
| server | string | Server's version string |
| cipher_alg | string | Encryption algorithm in use |
| mac_alg | string | Signing (MAC) algorithm in use |
| compression_alg | string | Compression algorithm in use |
| key_alg | string | Key exchange algorithm in use |
| host_key_alg | string | Server host key's algorithm |
| host_key | string | Server's key fingerprint |
| remote_location | record | Add geographic data related to remote host of connection |
| geo_location | record | Location |

ssl.log | SSL handshakes

| FIELD | TYPE | DESCRIPTION |
|---------------|--------|--|
| ts | time | Time when SSL connection first detected |
| uid & id | string | Underlying connection info - See conn.log |
| version | string | SSL/TLS version server chose |
| cipher | string | SSL/TLS cipher suite server chose |
| curve | string | Elliptic curve server chose when using ECDHE/CHE |
| server_name | string | Value of Server Name Indicator (SNI) extension |
| resumed | bool | Flag that indicates session was resumed |
| last_alert | string | Last alert seen during connection |
| next_protocol | string | Next protocol server chose using application layer next protocol extension, if present |
| established | bool | Flags if SSL session successfully established |
| ssl_history | string | SSL history showing which types of packets were received in which order. Client-side letters are capitalized, server-side lowercase. |

ssl_history

| | | | |
|---|----------------------|---|---------------------------|
| A | direction flipped | U | certificate_status |
| H | hello_request | A | supplemental_data |
| C | client_hello | Z | unassigned_handshake_type |
| S | server_hello | I | change_cipher_spec |
| V | hello_verify_request | B | heartbeat |
| N | new_session_ticket | D | application_data |
| X | certificate | E | end_of_early_data |
| K | server_key_exchange | O | encrypted_extensions |
| R | certificate_request | P | key_update |
| Y | server_hello_done | M | message_hash |
| N | certificate_verify | J | hello_retry_request |
| G | client_key_exchange | L | alert |
| F | finished | Q | unknown_content_type |
| W | certificate_url | | |

| | | |
|--|--------|---|
| cert_chain_fps | vector | All fingerprints for the certificates offered by the server |
| client_cert_chain_fps | vector | All fingerprints for the certificates offered by the client |
| subject | string | Subject of X.509 cert offered by server |
| issuer | string | Subject of signer of X.509 server cert |
| client_subject | string | Subject of X.509 cert offered by client |
| client_issuer | string | Subject of signer of client cert |
| sni_matches_cert | bool | Set to true if the hostname sent in the SNI matches the certificate, false if it does not. Unset if the client did not send an SNI. |
| request_client_certificate_authorities | vector | List of client certificate CAs accepted by the server |
| server_version | count | Numeric version of the server in the server hello |
| client_version | count | Numeric version of the client in the client hello |
| client_ciphers | vector | Ciphers that were offered by the client for the connection |
| sni_client_exts | vector | SSL client extensions |
| sni_server_exts | vector | SSL server extensions |
| ticket_lifetime_hint | count | Suggested ticket lifetime sent in the session ticket handshake by the server |
| dh_param_size | count | The diffie helman parameter size, when using DH |
| point_formats | vector | Supported elliptic curve point formats |
| client_curves | vector | The curves supported by the client |
| orig_alpn | vector | Application layer protocol negotiation extension sent by the client |
| client_supported_versions | vector | TLS 1.3 supported versions |
| server_supported_versions | count | TLS 1.3 supported version |
| psk_key_exchange_modes | vector | TLS 1.3 Pre-shared key exchange modes |
| client_key_share_groups | vector | Key share groups from client hello |
| server_key_share_group | count | Selected key share group from server hello |
| client_comp_methods | vector | Client supported compression methods |
| sigalgs | vector | Client supported signature algorithms |
| hashalgs | vector | Client supported hash algorithms |
| validation_status | string | Certificate validation result for this connection |
| ocsp_status | string | OCSP validation result for this connection |
| valid_ct_logs | count | Number of different logs for which valid SCTs encountered in connection |
| valid_ct_operators | count | Number of different log operators for which valid SCTs encountered in connection |

| | | |
|----------------------|--------|--|
| client_ciphers | vector | Ciphers that were offered by the client for the connection |
| sni_client_exts | vector | SSL client extensions |
| sni_server_exts | vector | SSL server extensions |
| ticket_lifetime_hint | count | Suggested ticket lifetime sent in the session ticket handshake by the server |
| dh_param_size | count | The diffie helman parameter size, when using DH |
| point_formats | vector | Supported elliptic curve point formats |
| client_curves | vector | The curves supported by the client |
| orig_alpn | vector | |