zeek® (ver_4.1.1)

Cheat Sheet v0.1

The Magic Unicorns

André Peuker, Lea Schertel, Richard Weiss
https://github.com/The-Magic-Unicorns

Zielsetzung

Netzwerk-Forensiker und -Analysten sind aktuell besonders mit modernen Angriffsszenarien konfrontiert und oftmals ist eine schnelle Reaktionszeit dieser Personengruppe unerlässlich. Dieser Cheat Sheet hat als Zielsetzung, gerade diese Personen bei Ihrer täglichen Arbeit zu unterstützen.

Disclaimer

Das Layout wurde an bekannte und oft verwendete Cheat Sheets angepasst, wenn auch nicht darauf ausgelegt, den zeek® Cheat Sheet zusammenzufalten.
Auch wurden existierende Cheat Sheets zum Thema zeek® betrachtet, aber nicht als Vorlage verwendet.
Auch wenn dieses Dokument im Rahmen einer Modul-Veranstaltung entstanden ist, freuen wir uns über Feedback, welches wir auch gerne einarbeiten werden. Danke!

Umgang mit diesem Cheat Sheet

Gerade während der Arbeit mit einem so enorm mächtigen Tool und während der Durchführung von Analysen ist es aus unserer Sicht unerlässlich, sich an die verschiedenen Möglichkeiten und Optionen zu erinnern. Dieses Dokument soll deswegen als Referenz für zeek® genutzt werden können. Alle beschriebenen Kommandos werden lokal auf dem System verwendet.

Happy Hunting!

1. Abschnitte und Übersicht

1. Die zeek® Tools
   - zeek
   - zeek-cut
   - zeekctl
   - zkg
2. Die zeek® Logs
   - conn.log
   - dns.log
   - http.log
   - files.log
   - ftp.log
   - ssl.log
   - x509.log
   - smtp.log
   - pe.log
   - dhcp.log
   - ntp.log
   - SMB Logs
     - DCE-RPC
     - Kerberos
     - NTLM
   - irc.log
   - rdp.log
   - traceroute.log
   - tunnel.log
   - dpd.log
   - known_*.log & software.log
   - weird.log & notics.log
   - capture_loss.log & reporter.log
3. zusätzliche Informationen
   - conn_state Information (conn.log)
   - history Information (conn.log)

Bei den zeek®-Tools werden nur die Arguments und Commands beschrieben, die sich nicht aus dem Kontext erschließen lassen.

Die zeek® Tools

zeek

Es handelt sich um ein kommandozeilen-basiertes Tool, um direkt für Analyseaktivitäten mit zeek zu interagieren; es ist für Live-Analysen und offline Arbeit gedacht.

Help:

- `zeek [options] [file ...]`
- `zeek --test [doctest-options] -- [options] [file ...]`

Optionen:

`-a | --parse-only`	Ausführungsende nach Skript-Parsern
`-b | --bare-mode`	Kein Laden von Scripts aus dem base/ Verzeichnis
`-i | --iface`	Lesen der Daten vom angegebenen Interface
`-f | --filter`	tcpdump Filter
`-e | --exec`	Ergänzung der Skripte um den angegebenen Code
`-p | --prefix`	Hinzufügen eines Präfixes zur Script-File Pfadauflösung (siehe Anmerkungen)
`-r | --readfile`	Einlesen der Daten aus der angegebenen tcpdump Datei
`-s | --rulefile`	Einlesen von Regeln aus der angegebenen Datei
`-t | --tracefile`	Aktivieren des Ausführungs-Tracing
`-w | --writefile`	Schreiben in eine angegebene tcpdump Datei
`-C | --no-checksums`	Ignorieren von Checksums
`-N | --print-plugins`	Anzeigen von verfügbaren Plugins
`-X | --zeekygen`	Erstellen der zeek Dokumentation anhand der Konfigurationsdatei

Beispiele:

- `zeek --help`
- `zeek -i en0 <list of scripts to load>`
- `zeek -r traffic.trace`
- `zeek -C -r traffic.trace`
- `zeek -r traffic.trace local`
- `zeek -r traffic.trace my-script.zeek`

- `mkdir output_directory; zeek -r traffic.trace LogAscii::logdir=output_directory`

Anmerkungen:

- Einige Verzeichnisse werden bei der Suche nach Skript-Dateien einbezogen:
  `./`
  `<prefix>/share/zeek/`
  `<prefix>/share/zeek/policy/`
  `<prefix>/share/zeek/site/`

  `zeek -r mypackets.trace frameworks/files/extract-all-files`

  Dies lädt das Skript:
  `$PREFIX/share/zeek/policy/frameworks/files/extract-all-files.zeek`

zeek-cut

Dieses Tool extrahiert die angegebenen Spalten aus den ASCII formatierten zeek Logs und präsentiert diese auf der Standardausgabe.

Help:

- `zeek-cut [options] [<columns>]`

Optionen:

`-c`	Schließt den ersten Block der Formatierungsbeschreibung in die Ausgabe ein
`-C`	Schließt alle Blöcke der Formatierungsbeschreibung in die Ausgabe ein
`-m`	wie `-c` nur in minimaler Darstellungsform
`-M`	wie `-C` nur in minimaler Darstellungsform
`-d`	Datum in lesbarer Repräsentation
`-D <format>`	Verwendet das angegebene `strftime` Datumformat
`-F <Trennzeichen>`	Verwendet das angegebene Trennzeichen als Feld-Trennzeichen (bspw. `-F ","`)
`-n`	Gibt alle Spalten abgesehen der angegebenen aus
`-u`	Wie `-d` nur in UTC
`-U <format>`	Wie `-D` nur in UTC

Beispiele:

- `zeek-cut --help`
- `cat conn.log | zeek-cut -m uid id.orig_h id.orig_p id.resp_h id.resp_p proto service duration conn_state`
- `cat conn.log | zeek-cut -m -u ts uid id.orig_h id.orig_p id.resp_h id.resp_p proto service duration conn_state`
- `cat conn.log | zeek-cut -F "," -m -u ts uid id.orig_h id.orig_p id.resp_h id.resp_p proto service duration conn_state`

zeekctl ZeekControl

Dieses Tool wird genutzt, um die zeek Installation zu verwalten. ZeekControl bietet zwei Funktionsweisen:

- traditionelles, Single-System-Setup
- Cluster-Modus

zeekctl besitzt einen interaktiven Modus, welcher durch zeekctl ohne weitere Parameter gestartet und durch `[zeekControl] >` dargestellt wird.

Help:

- `zeekctl`

Optionen:

`capstats [<nodes>] [<secs>]`	Anzeigen der Interface Statistiken mit capstats
`check`	Prüfen der Konfiguration vor der Installation
`config`	Anzeigen der zeekctl Konfiguration
`cron [--no-watch]`	Ausführen von Aufträgen, die über Cron laufen sollen
`cron enable|disable|?`	Aktivieren/Deaktivieren von cron-Jobs
`deploy`	`--check --install --restart`
`df [<nodes>]`	Anzeigen der Festplatten Auslastung
`diag [<nodes>]`	Anzeige der Node-Diagnose
`exec <shell cmd>`	Ausführen von shell Kommandos
`exit | quit`	Verlässt die interaktive Shell
`install`	Update der zeekctl Installation/Konfiguration
`netstats <nodes>`	Ausgabe der network counters
`nodes`	Ausgabe der Node Konfigurationen
`restart [--clean] [<nodes>]`	Stoppen und Neustart der Verarbeitung
`scripts [-c] [<nodes>]`	Anzeige der zeek Scripts, die von den Nodes geladen werden
`start [<nodes>]`	Starten der Verarbeitung
`status [<nodes>]`	Zusammenfassung der Node Status
`stop [<nodes>]`	Beenden der Verarbeitung
`top [<nodes>]`	Anzeige der zeek Prozesse, wie es von dem Linux Befehl `top` bekannt ist

zkg Zeek-Package-Manager

Mit zkg können die genutzten Packages verwaltet werden.

Help:

```
zkg
[-h] [--version] [--configfile FILE |
--user] [--verbose] [--extra-source
NAME=URL]
{test,install,bundle,unbundle,
remove,purge,refresh,upgrade,
load,unload,pin,unpin,list,
search,info,config,autoconfig,
env,create,template}
```

Beispiele:

- `zkg --help`
- `zkg list`
- `zkg search ja3`
- `zkg install`
- `zkg install zeek/salesforce/ja3`

Logs

conn.log

Info	https://docs.zeek.org/en/current/logs/conn.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/conn/main.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
proto	enum
service	string
duration	interval
orig_bytes	count
resp_bytes	count
conn_state	string
local_orig	bool
local_resp	bool
missed_bytes	count
history	string
orig_pkts	count
orig_ip_bytes	count
resp_pkts	count
resp_ip_bytes	count
tunnel_parents	set[string]
orig_l2_addr	string
resp_l2_addr	string
vlan	int
inner_vlan	int
speculative_service	string

dns.log

Info	https://docs.zeek.org/en/current/logs/dns.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/dns/main.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
proto	enum
trans_id	count
rtt	interval
query	string
qclass	count
qclass_name	string
qtype	count
qtype_name	string
rcode	count
rcode_name	string
AA	bool
TC	bool
RD	bool
RA	bool
Z	count
answers	vector[string]
TTLs	vector[interval]
rejected	bool
total_answers	count
total_replies	count
saw_query	bool
saw_reply	bool
auth	set[string]
addl	set[string]
original_query	string

http.log

Info	https://docs.zeek.org/en/current/logs/http.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/http/main.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
trans_depth	count
method	string
host	string
uri	string
referrer	string
version	string
user_agent	string
origin	string
request_body_len	count
response_body_len	count
status_code	count
status_msg	string
info_code	count
info_msg	string
tags	set[enum]
username	string
password	string
capture_password	bool
proxied	set[string]
range_request	bool
orig_fuids	vector[string]
orig_filenames	vector[string]
orig_mime_types	vector[string]
resp_fuids	vector[string]
resp_filenames	vector[string]
resp_mime_types	vector[string]
current_entity	HTTP::entity
orig_mime_depth	count
resp_mime_depth	count
client_header_names	vector[string]
server_header_names	vector[string]
omniture	bool
flash_version	string
cookie_vars	vector[string]
uri_vars	vector[string]

files.log

Info	https://docs.zeek.org/en/current/logs/http.html
Felder	https://docs.zeek.org/en/current/scripts/base/ frameworks/files/main.zeek.html

FELD	TYP Information
ts	time
fuid	string
tx_hosts	set[addr]
rx_hosts	set[addr]
conn_uids	set[string]
source	string
depth	count
analyzers	set[string]
mime_type	string
filename	string
duration	interval
local_orig	bool
is_orig	bool
seen_bytes	count
total_bytes	count

missing_bytes	count
overflow_bytes	count
timedout	bool
parent_fuid	string
md5	string
sha1	string
sha256	string
x509	X509::Info
extracted	string
extracted_cutoff	bool
extracted_size	count
entropy	double

ftp.log

Info	https://docs.zeek.org/en/current/logs/ftp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ftp/info.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
user	string
password	string
Command	string
arg	string
mime_type	string
file_size	count
reply_code	count
reply_msg	string
data_channel	FTP::ExpectedDataChannel
cwd	string
cmdarg	FTP::CmdArg
pending_commands	FTP::PendingCmds
passive	bool
capture_password	bool
fuid	string
last_auth_requested	string

ssl.log

Info	https://docs.zeek.org/en/current/logs/ssl.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ftp/info.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
version_num	count
version	string
cipher	string
curve	string
server_name	string
session_id	string
resumed	bool
client_ticket_empty_session_seen	bool
client_key_exchange_seen	bool
client_psk_seen	bool
last_alert	string
next_protocol	string
analyzer_id	count
established	bool
logged	bool
ssl_history	string
cert_chain	vector[Files::Info]
cert_chain_fps	vector[string]
client_cert_chain	vector[Files::Info]
client_cert_chain_fps	vector[string]
subject	string
issuer	string
client_subject	string
sni_matches_cert	bool
client_depth	count
always_raise_x509_events	bool
last_originator_heartbeat_request_size	count
last_responder_heartbeat_request_size	count
originator_heartbeats	count
responder_heartbeats	count
heartbleed_detected	bool
enc_appdata_packages	count
enc_appdata_bytes	count
server_version	count
client_version	count
client_ciphers	vector[count]
ssl_client_exts	vector[count]
ssl_server_exts	vector[count]
ticket_lifetime_hint	count
dh_param_size	count
point_formats	vector[count]
client_curves	vector[count]
orig_alpn	vector[string]
client_supported_versions	vector[count]
server_supported_version	count
psk_key_exchange_modes	vector[count]
client_key_share_groups	vector[count]
server_key_share_group	count
client_comp_methods	vector[count]
comp_method	count
sigalgs	vector[count]
hashalgs	vector[count]
validation_status	string
validation_code	int
valid_chain	vector[opaque[x509]]
ocsp_status	string
ocsp_response	string
valid_scts	count
invalid_scts	count
valid_ct_logs	count
valid_ct_operators	count
valid_ct_operators_list	set[string]
ct_proofs	vector[SSL::SctInfo]
notary	CertNotary::Response

x509.log

Info	https://docs.zeek.org/en/current/logs/x509.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/http/main.zeek.html

FELD	TYP Information
ts	time
fingerprint	string
certificate.version	count
certificate.serial	string
certificate.subject	string
certificate.issuer	string
certificate.not_valid_before	time
certificate.not_valid_after	time
certificate.key_alg	string
certificate.sig_alg	string
certificate.key_type	string
certificate.key_length	count
certificate.exponent	string
certificate.curve	string
extensions	vector[X509::Extension]
san.dns	vector[string]

san.uri	vector[string]
san.email	vector[string]
san.ip	vector[addr]
basic_constraints.ca	bool
basic_constraints.path_len	count
extension_cache	vector[any]
host_cert	bool
client_cert	bool
deduplication_index	X509::LogCertHash
always_raise_x509_events	bool
cert	string

smtp.log

Info	https://docs.zeek.org/en/current/logs/smtp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/smtp/main.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
trans_depth	count
helo	string
mailform	string
rcptto	set[string]
date	string
from	string
to	set[string]
cc	set[string]
reply_to	string
msg_id	string
in_reply_to	string
subject	string
x_originating_ip	addr
first_received	string
second_received	string
last_reply	string
path	vector[addr]
user_agent	string
tls	bool
process_received_from	bool
has_client_activity	bool
process_smtp_headers	bool
entity	SMT::Entity
fuids	vector[string]

is_webmail	bool

pe.log

Info	https://docs.zeek.org/en/current/logs/pe.html
Felder	https://docs.zeek.org/en/current/scripts /base/files/pe/main.zeek.html

FELD	TYP Information
ts	time
id	string
machine	string
compile_ts	time
os	string
subsystem	string
is_exe	bool
is_64bit	bool
uses_aslr	bool
uses_dep	bool
uses_code_integrity	bool
uses_seh	bool
has_import_table	bool
has_export_table	bool
has_cert_table	bool
has_debug_data	bool
section_names	Vector[strings]

dhcp.log

Info	https://docs.zeek.org/en/current/logs/dhcp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/dhcp/main.zeek.html

FELD	TYP Information
ts	time
uids	set[string]
client_addr	addr
server_addr	addr
client_port	port
server_port	port
mac	string
host_name	string
client_fqdn	string
domain	string
requested_addr	addr
assigned_addr	addr
lease_time	interval
client_message	string
server_message	string

msg_types	vector[string]
duration	interval
client_chaddr	string
msg_orig	vector[addr]
client_software	string
server_software	string
circuit_id	string
agent_remote_id	string
subscriber_id	string

ntp.log

Info	https://docs.zeek.org/en/current/logs/ntp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ntp/main.zeek.html

FELD	TYP Information
ts	time
uid	string
id	conn_id
version	count
mode	count
stratum	count
poll	interval
precision	interval
root_delay	interval
root_disp	interval
ref_id	string
ref_time	time
org_time	time
rec_time	time
xmt_time	time
num_exts	count

smb.log

Info	https://docs.zeek.org/en/current/logs/smb.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ntp/main.zeek.html

Wenn bei der Analyse die Verwendung des SMB Protokolls erkannt wird, werden ggf. folgende weitere Logs erzeugt:

- dce_rpc.log
- kerberos.log
- ntlm.log
- smb_cmd.log
- smb_files.log
- smb_mapping.log
- pe.log

san.uri	vector[string]
san.email	vector[string]
san.ip	vector[addr]
basic_constraints.ca	bool
basic_constraints.path_len	count
extension_cache	vector[any]
host_cert	bool
client_cert	bool
deduplication_index	X509::LogCertHash
always_raise_x509_events	bool
cert	string

smtp.log

Info	https://docs.zeek.org/en/current/logs/smtp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/smtp/main.zeek.html

FELD	TYP Information
ts	time
uid	string
(id.orig_h,	addr
id.orig_p,	port
id.resp_h,	addr
id.resp_p)	port
trans_depth	count
helo	string
mailform	string
rcptto	set[string]
date	string
from	string
to	set[string]
cc	set[string]
reply_to	string
msg_id	string
in_reply_to	string
subject	string
x_originating_ip	addr
first_received	string
second_received	string
last_reply	string
path	vector[addr]
user_agent	string
tls	bool
process_received_from	bool
has_client_activity	bool
process_smtp_headers	bool
entity	SMT::Entity
fuids	vector[string]

is_webmail	bool

pe.log

Info	https://docs.zeek.org/en/current/logs/pe.html
Felder	https://docs.zeek.org/en/current/scripts /base/files/pe/main.zeek.html

FELD	TYP Information
ts	time
id	string
machine	string
compile_ts	time
os	string
subsystem	string
is_exe	bool
is_64bit	bool
uses_aslr	bool
uses_dep	bool
uses_code_integrity	bool
uses_seh	bool
has_import_table	bool
has_export_table	bool
has_cert_table	bool
has_debug_data	bool
section_names	Vector[strings]

dhcp.log

Info	https://docs.zeek.org/en/current/logs/dhcp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/dhcp/main.zeek.html

FELD	TYP Information
ts	time
uids	set[string]
client_addr	addr
server_addr	addr
client_port	port
server_port	port
mac	string
host_name	string
client_fqdn	string
domain	string
requested_addr	addr
assigned_addr	addr
lease_time	interval
client_message	string
server_message	string

msg_types	vector[string]
duration	interval
client_chaddr	string
msg_orig	vector[addr]
client_software	string
server_software	string
circuit_id	string
agent_remote_id	string
subscriber_id	string

ntp.log

Info	https://docs.zeek.org/en/current/logs/ntp.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ntp/main.zeek.html

FELD	TYP Information
ts	time
uid	string
id	conn_id
version	count
mode	count
stratum	count
poll	interval
precision	interval
root_delay	interval
root_disp	interval
ref_id	string
ref_time	time
org_time	time
rec_time	time
xmt_time	time
num_exts	count

smb.log

Info	https://docs.zeek.org/en/current/logs/smb.html
Felder	https://docs.zeek.org/en/current/scripts /base/protocols/ntp/main.zeek.html

Wenn bei der Analyse die Verwendung des SMB Protokolls erkannt wird, werden ggf. folgende weitere Logs erzeugt:

- dce_rpc.log
- kerberos.log
- ntlm.log
- smb_cmd.log
- smb_files.log
- smb_mapping.log
- pe.log

Die aus unserer Sicht wichtigsten Logfiles wurden im Rahmen dieser Arbeit bearbeitet. weitere werden gerne im Nachgang noch hinzugefügt.

Zusätzliche Information

conn_state

S0	Verbindungsversuch, keine Antwort
S1	Verbindungsaufbau, nicht beendet
SF	Normaler Aufbau, normal beendet
REJ	Verbindungsversuch abgelehnt
S2	Verbindung aufgebaut und Versuch zum Ende von Dest/Responder
S3	Verbindung aufgebaut und Versuch zum Ende von Source/Originator
RSTO	Verbdinung aufgebaut, RST von Orig
RSTR	Responder sendet RST
RESTS0	Originator sendet SYN und RST, kein SYN-ACK vom Responder
RSTRH	Responder sendet SYN-ACK, gefolgt von RST, kein SYN vom Originator
SH	Originator sendet SYN, gefolgt von FIN, kein Responder SYN-ACK
SHR	Responder sendet SYN-ACK, gefolgt von FIN, kein Originator SYN
OTH	Kein SYN Packet gefunden, bspw. teilweise aufgezeichnete Verbindung

https://docs.zeek.org/en/current/scripts/base/protocols/conn/main.zeek.html

history

s	SYN ohne ACK bit
h	SYN+ACK
a	nur ein ACK
d	Packet mit Payload
f	Packet mit gesetztem FIN bit
r	Packet mit gesetztem RST bit
c	Packet mit fehlerhafter Checksum
g	Lücke im Inhalt (“content gap“)
t	Packet mit erneut übertragenem Inhalt
w	Packet mit zero windows advertisement
i	Packet Inkonsistenz
q	Packet mit Multi-Flags (bspw. SYN+FIN)
^	Verbindungsrichtung wurde geändert durch zeek-Heuristik

https://docs.zeek.org/en/current/scripts/base/protocols/conn/main.zeek.html