Cheatsheet – Scapy Functions and Classes

Important Info

This cheat sheet is neither complete nor does it contain all optional arguments for the listed functions. For more detailed information about the library, please consult the source code linked at the end of this document.

There have previously been problems when working with the "AH" layer due to scapy treating layers after "AH" as raw data. When in doubt check for the layer manually instead of calling a packet-level function.

It is always advisable to test your solutions with multiple test cases before handing them in!!!

Pcap

rdpcap(filename) .... Read a pcap or pcapng file and return a list of the included packets.

Packet

Packet.layers(self) ...... Returns a list of layer classes in this packet.
Packet.haslayer(self, cls) ... Returns "true" iff "self" has a layer that is an instance of "cls". Can alternatively be written as "cls in self".
Packet.getlayer(self, cls, nb=1) . Returns the "nb"th layer that is an instance of "cls". Similar behavior can be achieved by using "self"["cls"].
Packet.show(self) . Prints a hierarchical view of the packet. (Very useful for debugging!)

Ethernet

Ether()	Constructor and class name.
Ether.dst	Destination MAC address.
Ether.src	Source MAC address.
Ether.type	Ethernet Protocol.

Inet

IP()	Constructor and class name.
IP.version	IP version (always 4).
IP.ihl	Header length.
IP.tos	Type of service.
IP.len	Packet length.
IP.id	ID.
IP.flags	Flags.
IP.frag	Fragment Offset.
IP.ttl	Time to live.
IP.proto	Protocol.
IP.chksum	Checksum.
IP.src	Source IP address.
IP.dst	Destination IP address.

Authentication Header

AH()	Constructor and class name.
AH.nh	Next header.
AH.payloadlen	Payload length.
AH.reserved	Reserved.
AH.spi	Security parameter index.
AH.seq	Sequence number.
AH.icv	Integrity check value.
AH.padding	Padding.

Encapsulated Security Protocol

ESP()	Constructor and class name.
ESP.spi	Security parameter index.
ESP.seq	Sequence number.
ESP.data	Contains encrypted part and icv.

User Datagram Protocol

UDP()	Constructor and class name.
UDP.sport	Source port.
UDP.dport	Destination port.
UDP.len	Total length.
UDP.chksum	Checksum.

Transmission Control Protocol

TCP()	Constructor and class name.
TCP.sport	Source port.
TCP.dport	Destination port.
TCP.seq	Sequence number.
TCP.ack	Acknowledgment number.
TCP.dataofs	Data offset.
TCP.reserved	Reserved.
TCP.flags	Flags.
TCP.window	Window size.
TCP.chksum	Checksum.
TCP.urgptr	Urgent pointer.
TCP.options	Options.

Cheatsheet template created by Michael Müller, 2010
with extensive modifications by Velimir Gayevskiy, 2015

Cheatsheet – Scapy Functions and Classes

Transport Layer Security

TLS() Constructor and class name.

TLS.type Type of TLS message (for details see source linked at end of sheet).

TLS.version TLS version (for details see source linked at end of sheet).

TLS.len Ciphertext length.

TLS.iv IV.

TLS.msg Message.

TLS.mac Message Authentication Code.

TLS.pad Padding.

TLS.padlen Padding length.

TLS Handshake

Parent of all TLSHandshake classes.

_TLSHandshake.msgtype Type of handshake message (for details see source linked at end of sheet).

_TLSHandshake.msglen Message length.

_TLSHandshake.msg Message.

TLS Client Hello

TLSClientHello() Constructor and class name.

TLSClientHello.gmt_unix_time Time stamp.

TLSClientHello.random_bytes Random bytes.

TLSClientHello.sidlen Session ID length.

TLSClientHello.sid Session ID.

TLSClientHello.cipherslen Supported ciphers length.

TLSClientHello.ciphers Supported ciphers.

TLSClientHello.complen Supported compression algorithms length.

TLSClientHello.comp Supported compression algorithms.

TLSClientHello.extlen Extensions length.

TLSClientHello.ext Extensions.

TLS Server Hello

TLSServerHello() Constructor and class name.

TLSServerHello.gmt_unix_time Time stamp.

TLSServerHello.random_bytes Random bytes.

TLSServerHello.sidlen Session ID length.

TLSServerHello.sid Session ID.

TLSServerHello.cipher Chosen Cipher.

TLSServerHello.comp Chosen compression algorithm.

TLSServerHello.extlen Extensions length.

TLSServerHello.ext Extensions.

Domain Name System

DNS() Constructor and class name.

DNS.id ID.

DNS.qr Query/response.

DNS.opcode Opcode.

DNS.aa Authoritative answer.

DNS.tc Truncation.

DNS.rd Recursion desired.

DNS.ra Recursion available.

DNS.z Reserved zero.

DNS.ad Authentic data.

DNS.cd Checking disabled.

DNS.rcode Response Code.

DNS.qdcount Number of questions.

DNS.ancount Number of answer RRs.

DNS.nscount Number of authority RRs.

DNS.arcount Number of additional RRs.

DNS.qd Question list.

DNS.an Answer RR list.

DNS.ns Authority RR list.

DNS.ar Additional RR list.

Cheatsheet – Scapy Functions and Classes

DNS Resource Record

DNSRR() .... Constructor and class name.
DNSRR.rrname .... Domain name.
DNSRR.type ...... Record type (for details see source linked at end of sheet).
DNSRR.cacheflush .... Cache-flush.
DNSRR.rclass ..... Record class (for details see source linked at end of sheet).
DNSRR.ttl .... Time to live.
DNSRR.rdlen .... Record data length.
DNSRR.rdata .... Record data.

DNS RRSIG Resource Record

DNSRRRSIG() .. Constructor and class name.
DNSRRRSIG.typecovered Record type being signed.
DNSRRRSIG.algorithm Signature algorithm.
DNSRRRSIG.labels .... Number of labels.
DNSRRRSIG.originalttl ...... Original TTL.
DNSRRRSIG.expiration .... Expiration date.
DNSRRRSIG.inception ...... Inception date.
DNSRRRSIG.keytag .... Tag of key used.
DNSRRRSIG.signersname .. Name of signer.
DNSRRRSIG.signature .... Signature.

DNS DNSKEY Resource Record

DNSRRDNSKEY() .... Constructor and class name.
DNSRRDNSKEY.flags .... Flags.
DNSRRDNSKEY.protocol .... Protocol.
DNSRRDNSKEY.algorithm .... Signature algorithm.
DNSRRDNSKEY.publickey ..... Public key.

DNS DS Resource Record

DNSRRDS() ..... Constructor and class name.
DNSRRDS.keytag .... Key ID.
DNSRRDS.algorithm ... Signature algorithm.
DNSRRDS.digesttype .... Type of hash.
DNSRRDS.digest .... Hash value of key.

Combining Layers

Layer objects can be concatenated using the "/" operator to form packets containing both. E.g., write "IP()/UDP()" to combine instances of the IP and UDP layers into one packet.

Source Code

Pcap:
`https://github.com/secdev/scapy/blob/master/scapy/utils.py#L1336`
Packet:
`https://github.com/secdev/scapy/blob/master/scapy/packet.py#L82`
Ethernet:
`https://github.com/secdev/scapy/blob/master/scapy/layers/l2.py#L281`
Inet:
`https://github.com/secdev/scapy/blob/master/scapy/layers/inet.py#L533`
AH:
`https://github.com/secdev/scapy/blob/master/scapy/layers/ipsec.py#L71`
ESP:
`https://github.com/secdev/scapy/blob/master/scapy/layers/ipsec.py#L119`
UDP:
`https://github.com/secdev/scapy/blob/master/scapy/layers/inet.py#L818`
TCP:
`https://github.com/secdev/scapy/blob/master/scapy/layers/inet.py#L737`
TLS:
`https://github.com/secdev/scapy/blob/master/scapy/layers/tls/record.py#L234`
TLS type and version dictionaries:
`https://github.com/secdev/scapy/blob/master/scapy/layers/tls/basefields.py#L16`
TLS handshake class and type dictionary:
`https://github.com/secdev/scapy/blob/master/scapy/layers/tls/handshake.py#L92`
TLS Client Hello:
`https://github.com/secdev/scapy/blob/master/scapy/layers/tls/handshake.py#L253`
TLS Server Hello:
`https://github.com/secdev/scapy/blob/master/scapy/layers/tls/handshake.py#L487`
DNS:
`https://github.com/secdev/scapy/blob/master/scapy/layers/dns.py#L1272`
DNS RR:
`https://github.com/secdev/scapy/blob/master/scapy/layers/dns.py#L1166`
DNS RRSIG RR:
`https://github.com/secdev/scapy/blob/master/scapy/layers/dns.py#L860`
DNS DNSKEY RR:
`https://github.com/secdev/scapy/blob/master/scapy/layers/dns.py#L893`
DNS DS RR:
`https://github.com/secdev/scapy/blob/master/scapy/layers/dns.py#L910`