DDOS RESILIENCY SCORE (DRS)

"An open standard for quantifying an
Organization's resiliency to withstand DDoS attacks"

Version 2.00.00
28 October 2023

.....

1. Introduction

The DDoS Resiliency Score, or in short 'DRS', is value denoting the ability of an organization to withstand various degrees of DDoS attacks.
The score is exponential, like the Richter scale for earthquakes). The exponential measurement reflects the large variation of DDoS attacks and enables placing on the same scale very simple, low-volume attacks together with sophisticated, multi-vector, 100Gbps attacks.

The DRS scoring mechanism is based on seven ascending levels of DDoS attacks and the ability to successfully withstand each of them. Each level introduces new types of attacks, more sophisticated attack vectors, and a larger volumes of traffic. Similarly, the requirements on the defending side increase, with each level requiring shorter mitigation response time and smaller latency.

Achieving a score of 3.8, for example, would mean that an organization passed the 3rd level of attacks, but failed with some of the 4th level of attacks.

Due to pragmatic reasons, the highest score is currently set to 7. Possibly, the scale may grow in the future to respond to developments of new DDoS attacks.

2. Usage

The DRS provides a measurement tool, allowing organizations to evaluate in quantitative terms their mitigation strategy and ability to withstand DDoS attacks. The DRS also introduces objectiveness into a field of much debate. For example, it enables comparing the effectiveness of different technologies by assigning a score to each one. Last but not least, the DRS introduces a common language in which management and technical teams can communicate. A score of 4.7 can

indicate to management that the mitigation capabilities have improved since the previous score of 3.5, while it also encapsulate a list of specific attack vectors that will and will not be blocked, which the technical teams can analyze. It is furthermore recommended that each DDoS related decision such as technology investment is measured using DRS to ensure cost efficiency.

The DRS can also be used as a framework, and participating parties are encouraged to introduce derived works that reflect developments in industries or geographic regions. All derivatives that fall under these conditions MUST be public available. As derivatives are only such works allowed, that extend and refine the DRS, but shall not contradict its definitions.

3. Phases Definition

3.1 Phases Abstract Definition

The following is a high level definition of the phases. There are 7 phases. Each phase also has a nick name. In this section each phase is abstractly defined.

PHASE 1 ("poking") - A basic poking attack checks if there is any DDoS resiliency whatsoever. Only 2 vectors at low rate are included here.

PHASE 2 ("script kiddy") - A primitive "script kiddy" DDoS attack. UDP Flood is added. Attack rate slightly increase but is still low.

LEVEL 3 ("basic") - A "basic" DDoS attack with multiple attack vectors. Includes more bandwidth, but not yet sophisticated.

LEVEL 4 ("sophisticated") – This level is the first to include sophisticated attack vectors. For example, UDP Amplified Reflected attacks start at this phase.

LEVEL 5 ("persistent") - Includes persistent attacks: multi-vector, using even more sophisticated attack vectors, and looking for weakness while also increasing in volume. Similar to an Advanced Persistent Attack (APT) attack.

LEVEL 6 ("extreme") - An extreme DDoS attack. Sophistication and volume increase and includes exotic attacks.

LEVEL 7 ("state sponsored") - This level uses all

known techniques to break the DDoS defense.

3.2 Considered Resiliency factors

The resiliency of an organization to DDoS attacks is defined by multiple factors listed below. Each of these factors is increased with each level.

Attack vector types – each level introduces more attack vectors.

Attack vector volume - each level includes attack vector with higher volume. Volume includes mere bandwidth (bits per seconds), packets per second and transaction per second. Attack volume boundaries are defined in section 3.3.

Attack vector sophistication – each level introduces more sophisticated attack vectors. This is defined in section 3.4.

Mitigation requirements – each level requires the organization to mitigate the attack more effectively, measuring parameters such as mitigation response time and latency during mitigation. This is defined in section 3.5.

3.3. Maximal volume per attack level

The following section describes the maximal attack volumes that will be used at each level. Volumes are provided in BPS (bits-per-second), PPS (packets-per-second) and TPS (transactions-per-second). Note that each attack vector may not utilize the maximal volume defined for that level.

Level	Maximal volume per phase		
	BPS	PPS	TPS
1	100 Mbps	25K	500
2	1 Gbps	250K	5 K
3	10 Gbps	2M	50 K
4	100 Gbps	25M	200 K
5	500 Gbps	125M	1 M
6	1 Tbps	250M	5 M
7	5 Tbps	1B	25 M

3.4 Attack Vector Sophistication Properties

In each level, attacks become more advanced not only in their sheer size or type of attack vectors, but also in the properties of each attack. For example, IP Address Spoofing is a technique used in DDoS to generate more effective attacks. Spoofing and other techniques used to create more effective attacks will be referred as 'Sophistication Properties'. Loosely, Sophisticated Properties' are the equivalent of 'evasion techniques' used non-DDoS attacks.
The following section describes each property, states in which phase it is first introduced ("Start at Phase"), and to which attack vectors it is applicable ("Applicable to").

3.4.1 IP Address Spoofing

IP Address Spoofing (in short 'Spoofing') is the creation of Internet Protocol (IP) packets with a forged source IP address.

Property Name	Spoofing
Starts at level	2
Applicable to	Stateless attacks

3.4.2 URL Randomization

URL Randomization is a technique used to produce a more effective DDoS attack, which can bypass some mitigation technologies as well as caching-based protection methods. It is used in web based attack, HTTP and HTTPS. Randomization can either be done in the Path or Parameters or both.

Property Name	URL Randomization

Starts at level	4
Applicable to	HTTP and HTTPS

3.4.3 Hiding the Attack Tools Fingerprints

Many tools used for attacks leave fingerprints in the attacking packets. For example, the headless-browser PhantomJS states by default its name in the User-Agent field. This allows mitigation technologies to block the attack using a signature. However, sophisticated attackers will strive to hide their attack tools fingerprints that are not essential.

Property Name	Hide Attack Tool Fingerprint
Starts at level	6
Applicable to	Attack vectors original from tools that have fingerprints

3.4.4. Attacking the network directly

DDoS mitigation often uses an architecture in which a CDN or large reverse proxies are placed in front of the web services as a protection layer. However, sophisticated attackers will attempt to reveal the origin network or IP address and attack directly, making the mitigation layer completely useless.
When this sophistication layer is enabled the attack should strive first to reveal the origin network or IP addresses and attack them directly. In case they origin is not found or the attack is ineffective the attack will as normal (targeting the domain name).

Property Name	Attacking the network directly
Starts at level	6
Applicable to	All attacks

3.5 Mitigation Requirements per Level

Resiliency is also a factor of the defending entity. An organization that is able to fully mitigate an attack after ten seconds is more resilient than one that can mitigate the same attack after ten minutes. This parameter is referred to as 'Mitigation Response Time'.

Another parameter is ‘Latency’. A service that under an on-going attack has 1 millisecond extra latency is more resilient than a service that suffers from an extra 1 second latency.

Both Mitigation Response Time and Latency are inserted into the score in a similar manner. Each level has growing requirements. An attack vector will be considered passed if all attacked service quickly become functional and with reasonable latency.

3.5.1 Mitigation Response Time requirements

Mitigation Response Time for each level is defined as follows:

Level	Maximal Outage
1	6 hours
2	4 hours
3	1 hours
4	10 minutes
5	5 minutes
6	1 minutes
7	20 seconds

3.5.1 Latency requirements

Latency is defined as the delta or extra rime in the roundtrip an average packet in the service. The delta is

in comparison to the normal roundtrip time not under attack.

Level	Maximal latency
1	10 seconds
2	5 seconds
3	3 seconds
4	2 seconds
5	1.5 seconds
6	1 seconds
7	0.5 seconds

3.6. Magnitude of Bots (nodes) per Phase

The intensity of a DDoS attack is not solely based on the volume or type of traffic but also on the sheer number of devices, or bots, participating in the attack. Sophisticated attackers have access to vast networks of compromised devices, enabling them to launch significant attacks. To fully understand and prepare for potential attacks, it is essential to consider the number of bots that might be used at each level of an attack.

Level	Maximal bots count
1	10 bots
2	50 bots
3	500 bots

4	1,000 bots
5	5,000 bots
6	50,000 bots
7	1M bots

3.7. Attack Campaign Persistence per Phase

The duration of an attack can reflect its complexity and the tenacity of the attackers. Longer, more persistent attacks generally require a more concerted effort and advanced resources. The following table specifies the maximum length of time each attack full ongoing campaign is expected to persist per level:

Level	Persistence Duration
1	1 hour
2	12 hours
3	24 hours
4	7 days
5	30 days
6	6 months
7	1 year

4. Attack Vectors

The following section defines the attack vectors that are used in each level. At first the Attack Vector types are defines. 'SYN Floods' and 'UDP Flood' are examples of Attack Vector Types. Then the Attack Vectors are defines which are based on the Attack Vector Types but also

include rates and sophistication properties. For example, '111001 SYN Flood : vol_pps= 5K' and '211001 SYN Flood : vol_pps= 50K, ip_spoofing' represents two different Attack Vectors, that are based on the same type.

4.1 Attack Vector Notation

Each attack vector will be specified in the following format

<ID> <Attack Vector Name> <Properties>

4.1.1 Attack Vector ID (ID)

The Attack Vector ID, specified in short as 'ID', is a unique number representing the attack vector. The ID is a 6-digit number with the following format:

PFANNN

Where each digit, represented by a letter stands for

Digit	Definition
P	The level of the attack vector
F	Attack vector family
A	The first level this attack vector appears
NNN	Unique number assign to each Attack Vector Type

'FANNN' also represents a unique Attack Vector Type identifier and when 'P' is prepended it represent a unique Attack Vector identifier.

4.1.1.1. Attack Vector Family Enumeration

The following table defines the Attack Vector Family enumeration which is used as part of the Attack Vector Type.

Digit	Attack Vector Family
1	Network Attacks - TCP
2	Network Attacks – UDP
3	Network Attacks – Other(e.g. ICMP)
5	Application Attacks
8	Low-and-Slow

4.1.2. Attack Vector Types

The name of the attack vector (as defined in section 4.2)

4.1.3. Attack Vector Properties

Throughout the different levels the same attack vectors are used. For example, SYN Flood will be used at all levels, but each time its intensity and sophisticated is increased. The intensity of SYN Flood in Level 1 is 10K PPS, in Level 2 it is 100K PPS, and so on.

4.1.4.1 Attack Vector Properties - Volume

Property Notation	Description
vol_bps=VALUE	attack volume in bits-per-second as define by field VALUE.
vol_pps=VALUE	attack volume in packets-per-second as define by field VALUE.
vol_cps=VALUE	attack volume in connection-per--second as define by field VALUE.
vol_tps=VALUE	attack volume in transactions-per--second (AKA request-per-seconds) as define by field VALUE.

VALUE will be specified as numeric value commonly with 'K', 'M', 'G' representing 'Kilo', 'Mega' and 'Giga' respectively.

4.1.4.1 Attack Vector Properties – Sophistication Properties

The attack vectors Sophistication Properties are defined in section 3.4. This section defines the notation for each one.

Property Notation	Sophisticate Property (section specified)
ip_spoofing	IP Address Spoofing (3.4.1)
url_rand	URL Randomization (3.4.2)

no_fingerprint	Hiding Attack Vector Fingerprint (3.4.3)
direct_ip	Attacking the network directly (3.4.4)

4.2. Attack Vectors Types Specification

The following section specifies the attack vectors by order of appearance in the different Levels. The names used are industry acceptable names and additional information about attack vectors can be found on the web.

The attacks are specified in the following format

<ID> <name>
<Description & specification>
Newline

11001 SYN Flood
A flood of TCP SYN packets, data size SHOULD be 0.

51002 HTTP GET Flood
A flood of HTTP request.

22003 UDP Flood
A flood of UDP packets. Data size should be large or even maximal. DST port May be 80.

13004 TCP RST Flood
A flood of RST packets flood.

33005 ICMP Flood
A Flood of ICMP ping packets. Data size SHOULD be large.

53006 HTTPS GET Flood
A flood of HTTPS request.

14007 TCP SYN+ACK Flood
A flood of SYN+ACK packets. The data size SHOULD be small or zero.

14008 TCP ACK Flood
A flood of ACK packets. The data size SHOULD be small or zero.

14009 TCP PSH Flood
A flood of TCP PSH packets.

14010 TCP FIN Flood
A flood of TCP FIN packets. The data size SHOULD be small or zero.

24011 NTP Reflection Flood
An NTP Reflected flood the using MONLIST argument

54012 DNS Query Flood
A flood of DNS
queries.

25013 DNS Garbage Flood
A flood of UDP packets. DST Port must be 53. The data is garbage (not proper DNS request or reply). Data size SHOULD be large.

55014 HTTP Flood Cookie Support
An HTTP Flood in which the attack tool is able to support cookie and respond to an HTTP 302 Redirect response.

85015 HTTP Search Page
An HTTP flood targeted at one or more search functions in the attacked website.

85016 HTTP Large File Download
An HTTP flood targeted at one or more large files located at the site.

55017 DNS Recursive
A flood on DNS packet in which the subdomain is ever changing (1000.ddostarget.com, 1001.ddostarget.com, etc)

55018 Slow Post (HTTP)
Slow Post is an HTTP based low-and-slow DDoS attack using POST request with large Content-Length, however the attacker send the data byte-by-byte keep the connection ever open. R.U.D.Y. was the first tool that introduced this type of attack.

55019 Slowloris
A low-and-slow HTTP based attack against Apache server family. The attack sends multiple HTTP request in which each request is incomplete.

55020 SSL Renegotiation

A low-and-slow HTTPS based attack. Using the SSL-Renegotiation option the attack causes the server to renegotiate the SSL that consumes large compute power.

16021 Tsunami SYN Flood
SYN Flood in which the data size is very large (normally there is no data in SYN packets)

26022 CHARGEN Reflective Flood
A type of UDP Reflection Amplification attack using the CHARGEN protocol

56023 HTTP Flood JavaScript Support
An HTTP flood in the attacking client is able to process JavaScript (JS) and therefore pass standard JS DDoS mitigation challenges.

56024 HTTPS Flood Cookie Support
As as '55014 HTTP Flood Cookie Support' but over the HTTPS.

57025 HTTP Flood Headless Browser
An HTTP flood in the attacking client is a headless browser and therefore encompasses all the technologies and libraries of a normal browser and can pass multiple standard DDoS mitigation challenges.

57026 HTTPS Flood JavaScript Support
Same as 56023 HTTP Flood JavaScript Support' but over the HTTPS.

57027 HTTPS Flood Headless Browser
Same as '57025 HTTP Flood Headless Browser' but over the HTTPS protocol.

57028 Slow Post (HTTPS)
Same as '55018 Slow Post (HTTP)' but over the HTTPS protocol.

4.3 Attack Vector per Level Specification

The following attack vectors are included in each of the levels. The format used will be

<ID> <Attack Vector name> : <Attack Vector properties >

LEVEL 1

111001 SYN Flood : vol_pps= 25K

151002 HTTP GET Flood : vol_tps= 500

LEVEL 2

211001 SYN Flood : vol_pps= 250K, ip_spoofing

251002 HTTP GET Flood : vol_tps= 5K

222003 UDP Flood : vol_bps= 1G, ip_spoofing

LEVEL 3

311001 SYN Flood : vol_pps= 2M, ip_spoofing

351002 HTTP GET Flood : vol_tps= 50K

322003 UDP Flood : vol_bps= 10G, ip_spoofing

313004 TCP RST Flood : vol_bps= 10G, ip_spoofing

333005 ICMP Flood : vol_bps= 10G, ip_spoofing

353006 HTTPS GET Flood : vol_tps= 50K

LEVEL 4

411001 SYN Flood : vol_pps= 25M, ip_spoofing

451002 HTTP GET Flood : vol_tps= 200K, url_rand

422003 UDP Flood : vol_bps= 100G, ip_spoofing

413004 TCP RST Flood : vol_bps= 100G, ip_spoofing

433005 ICMP Flood : vol_bps= 100G, ip_spoofing

453006 HTTPS GET Flood : vol_tps= 200K, url_rand

414007 TCP SYN+ACK Flood : vol_bps= 100G, ip_spoofing

414008 TCP ACK Flood : vol_bps= 100G, ip_spoofing

414009 TCP PSH Flood : vol_bps= 100G, ip_spoofing

414010 TCP FIN Flood : vol_bps= 100G, ip_spoofing

424011 NTP Reflection Flood : vol_bps= 100G

424012 DNS Query Flood : vol_tps= 200K, ip_spoofing

LEVEL 5

511001 SYN Flood : vol_pps= 125M, ip_spoofing

551002 HTTP GET Flood : vol_tps= 1M, url_rand

522003 UDP Flood : vol_bps= 500G, ip_spoofing

513004 TCP RST Flood : vol_bps= 500G, ip_spoofing

533005 ICMP Flood : vol_bps= 500G, ip_spoofing

553006 HTTPS GET Flood : vol_tps= 1M, url_rand

514007 TCP SYN+ACK Flood : vol_bps= 500G, ip_spoofing

514008 TCP ACK Flood : vol_bps= 500G, ip_spoofing

514009 TCP PSH Flood : vol_bps= 500G, ip_spoofing

514010 TCP FIN Flood : vol_bps= 500G, ip_spoofing

524011 NTP Reflection Flood : vol_bps= 500G

524012 DNS Query Flood : vol_tps= 1M, ip_spoofing

525013 DNS Garbage Flood : vol_bps= 500G, ip_spoofing

555014 HTTP Flood Cookie Support : vol_tps= 1M,
url_rand

585015 HTTP Search Page : vol_tps= 1M

585016 HTTP Large File Download : vol_tps= 1M

525017 DNS Recursive : vol_tps= 1M, ip_spoofing

555018 RUDY (HTTP) : vol_tps= 1M

555019 Slowloris : vol_tps= 1M

555020 SSL Renegotiation : vol_tps= 1M

LEVEL 6

611001 SYN Flood : vol_pps=250M, ip_spoofing, direct_ip

651002 HTTP GET Flood : vol_tps=5M, url_rand, direct_ip

622003 UDP Flood : vol_bps=1T, ip_spoofing, direct_ip

613004 TCP RST Flood : vol_bps=1T, ip_spoofing,
direct_ip

633005 ICMP Flood : vol_bps=1T, ip_spoofing, direct_ip

653006 HTTPS GET Flood : vol_tps=5M, url_rand,
direct_ip

614007 TCP SYN+ACK Flood : vol_bps=1T, ip_spoofing,
direct_ip

614008 TCP ACK Flood : vol_bps=1T, ip_spoofing,
direct_ip

614009 TCP PSH Flood : vol_bps=1T, ip_spoofing,

direct_ip

614010 TCP FIN Flood : vol_bps=1T, ip_spoofing,
direct_ip

624011 NTP Reflection Flood : vol_bps=1T, direct_ip

654012 DNS Query Flood : vol_tps=5M, ip_spoofing,
direct_ip

625013 DNS Garbage Flood : vol_bps=1T, ip_spoofing,
direct_ip

655014 HTTP Flood Cookie Support : vol_tps=5M,
url_rand, direct_ip

685015 HTTP Search Page : vol_tps=5M, direct_ip

685016 HTTP Large File Download : vol_tps=5M, direct_ip

655017 DNS Recursive : vol_tps=5M, ip_spoofing,
direct_ip

655018 Slow Post (HTTP) : vol_tps=5M ,no_fingerprint,
direct_ip

655019 Slowloris : vol_tps=5M ,no_fingerprint,
direct_ip

655020 SSL Renegotiation : vol_tps=5M, direct_ip

616021 Tsunami SYN Flood : vol_bps=1T, ip_spoofing,
direct_ip

626022 CHARGEN Reflective Flood : vol_bps=1T, direct_ip

656023 HTTP Flood JavaScript Support : vol_tps=5M,
url_rand, direct_ip

656024 HTTPS Flood Cookie Support : vol_tps=5M,
url_rand, direct_ip

LEVEL 7

711001 SYN Flood : vol_pps=1B, ip_spoofing, direct_ip

751002 HTTP GET Flood : vol_tps=25M, url_rand,
direct_ip

722003 UDP Flood : vol_bps=5T, ip_spoofing, direct_ip

713004 TCP RST Flood : vol_bps=5T, ip_spoofing,

direct_ip

733005 ICMP Flood : vol_bps=5T, ip_spoofing, direct_ip

753006 HTTPS GET Flood : vol_tps=25M, url_rand,
direct_ip

714007 TCP SYN+ACK Flood : vol_bps=5T, ip_spoofing,
direct_ip

714008 TCP ACK Flood : vol_bps=5T, ip_spoofing,
direct_ip

714009 TCP PSH Flood : vol_bps=5T, ip_spoofing,
direct_ip

714010 TCP FIN Flood : vol_bps=5T, ip_spoofing,
direct_ip

724011 NTP Reflection Flood : vol_bps=5T, direct_ip

754012 DNS Query Flood : vol_tps=25M, ip_spoofing,
direct_ip

725013 DNS Garbage Flood : vol_bps=5T, ip_spoofing,
direct_ip

755014 HTTP Flood Cookie Support : vol_tps=25M,
url_rand, direct_ip

785015 HTTP Search Page : vol_tps=25M, direct_ip

785016 HTTP Large File Download : vol_tps=25M,
direct_ip

755017 DNS Recursive : vol_tps=25M, ip_spoofing,
direct_ip

755018 Slow Post (HTTP) : vol_tps=25M ,no_fingerprint,
direct_ip

755019 Slowloris : vol_tps=25M ,no_fingerprint,
direct_ip

755020 SSL Renegotiation : vol_tps=25M, direct_ip

716021 Tsunami SYN Flood : vol_bps=5T, ip_spoofing,
direct_ip

726022 CHARGEN Reflective Flood : vol_bps=5T, direct_ip

756023 HTTP Flood JavaScript Support : vol_tps=25M,

url_rand, direct_ip

756024 HTTPS Flood Cookie Support : vol_tps=25M,

url_rand, direct_ip

757025 HTTP Flood Headless Browser : vol_tps=25M,

url_rand, no_fingerprint, direct_ip

757026 HTTPS Flood JavaScript Support : vol_tps=25M,

url_rand, direct_ip

757027 HTTPS Flood Headless Browser : vol_tps=25M,

url_rand, no_fingerprint, direct_ip

757028 Slow Post (HTTPS) : vol_tps=25M ,no_fingerprint,

direct_ip

5. Score Calculation Procedure

The following section explains how the actual score is calculated.

5.1 General Description

The DRS score is measured using level based testing. To pass a level, the protection measures used by an organization are expected to simultaneously mitigate the vectors used in that level. Attacks are run sequentially: level 1 attack vectors, level 2 attack vectors, and so on. If the organization is able to withstand the attack of that level, it passes to the next one. For example, if the organization was able to withstand the attack vectors included in Level 1 the test continues to Level 2 attack

vectors. This process continues until the organization fails a certain level.

5.2 Passing or Failing an Attack Vector

If the organization was able to withstand an attack vector and effectively provide its services in a timely manner, then the attack vector is considered to be passed ('Passed Attack Vector'), otherwise it is defined as failed ('Failed Attack Vector').

5.3 Level Score

After running the attacks of a given level and collecting results, the level score is calculated. This value is referred to as 'Level Score'.
Level Score is calculated as the Passed Attack Vector divided by the total number of vectors in that level, plus the level number minus one.
For example, if in level 3, 4 attack vectors were passed out of 10 (failing that level), then the Level Score will be '2.4'.

5.4 Passing, Meeting and Failing a Level

The Level Score determines if that level was passed, met or failed. This depends on the 'Passing Score' and 'Failing Score' per level defined in the table below. If the Level Score is above the Passing Score the level is considered as 'Passed' and the test will continue to the next level. If the Test Score is below the Failing Score the level is considered as 'Failed'. If the Level Score is in between the two, the Level is considered as 'Met'. While both Failed and Met do not entitle the test to continue to the next phase they effect the Final Score as defined below.

Level	Passing Score	Failing Score
1	75%	40%
2	75%	40%
3	75%	40%
4	85%	40%
5	85%	40%
6	85%	40%
7	85%	40%

5.5 Final Score

For each test that is passed the test continue to the next level. If the last score in the last Level was a Met score than that score is the 'Final Score'. If the last score was a Failed score than then the Final Score is the previous Phase Score, i.e. the last passed phase.

6. Disclaimer

The DDoS Resiliency Score (DRS) was developed by Red Button Ltd. as a practical tool for evaluating an organization's mitigation strategy and ability to withstand DDoS attacks. However, the developer of DRS does not provide an assurance or any legal warranty as to

the ability of DRS to fully prevent an attack on an organization, whether DDoS attack or other. Red Button Ltd. hereby disclaims any other warranty, expressed or implied, including, without limitation, any warranty or fitness of DRS for a particular purpose.

7. Legal Notice