Towards density metrics for Internet health

L. Aaron Kaplan[1] Éireann Leverett[2]

March 20, 2017
Version: 1.2

[1]CERT.at: kaplan@CERT.at
[2]Privacy International: eireann@privacyinternational.org

Abstract

We explore possibilities for density based metrics for measuring 'Cyber health'. Previous work[1], focused on the *risk to others*. That is, the sum of the potential impact of systems which are vulnerable to UDP amplification attacks, emphon to others. This previous work resulted in a workable metric for ASNs. We called this score 'metric version 2.1'. Version 2.1 has its merits - for example with a slight variation of it, we can calculate a 'DDoS potential' per country or ASN, or indeed globally. However, the CyberGreen stats working group identified the wish to compare risk in a normalized way – 'density metrics'.

As of today, no clear 'density metric' has been decided upon yet. This paper tries to compare different approaches and weighs them against each other.

We start this paper with a brief recap of metric version 2.1 and show its successful applications. We then continue to explore different candidates for normalization by comparing the pros and cons.

Based on the analysis of the candidates we will propose a new metric version 2.2 – to be seen as one in a series of multiple metrics.

Finally, we conclude with a section on future work.

[1] `http://www.cybergreen.net/img/medialibrary/CyberGreen%20Metrics%20v.2.pdf`

Terms and definitions

Vulnerable systems : in our context, we mean devices which are vulnerable to be misused in a UDP amplification attack.

Infected systems : also known as hacked systems, compromised systems. These might be used as well (as part of a botnet) for a DDOS attack. However, currently in this paper we will ignore them.

Risk either an infected systems or vulnerable systems. Please note that the definition of 'risk' in this paper differs from the risk-assement or -management language.

Cyber Health the absence of (vulnerabilities and/or infected systems) in the Internet or parts of it.

Metric synonymous with score in our paper. A function of the number of vulnerable or infected devices.

Normalization / density used interchangeably for a constant or a function by which we normalise the number of vulnerable or infected devices for a given ASN or country with another value of interest E.G. machine population, human population, or GDP.

System, device, machine, service we use these terms interchangeably - all meaning some networked device with a public IP address. Note that in this definition, a system might actually hide multiple devices behind a NAT. CyberGreen will not be able to measure these hidden devices at the moment.

ASN Autonomous System Number - a unique ID assigned to an Autonomous System.

BGP Border Gateway Protocol

DDOS Denial of Service attack. Specifically, in this paper when we talk about DDoS attacks, we implicitly will only talk about UDP based amplification attacks. At the time of this writing, CyberGreen only processes open data on the following UDP based amplification risks:

- open recursive DNS
- open NTP devices
- open SNMP devices
- open SSDP devices

See also Cybergreen's list of processed risks: `https://github.com/cybergreen-net/reference-data/blob/master/risks/risks.csv`. By 'open' we mean vulnerable to being misused in an UDP amplification attack.

RIR Regional Internet Registry (example: RIPE).

CyberGreen metrics version 2.1

'CyberGreen Metrics'[2] defines the current version 2.1 metrics.

It is given as

$$CG_i = rank\left(\sum_{j=1}^{n} count_{i,j} * AF(risk_j)\right)$$

Where $n = 1...4$ (for the four risks "Open NTP", "Open recursive DNS", "Open SNMP" and "OpenSSDP"). CG_i is the CyberGreen index of $country_i$ or ASN_i.

The way the current `stats.cybergreen.net` page was implemented[3], the rank was omitted:

For an ASN_i, the combined risk (consisting of n individual risks is given by the weighted sum:

$$risk_{ASN_i} = \sum_{j=1}^{n} risk_j(ASN_i) * AF(risk_j)$$

$AF(risk_j)$ is the Amplification Factor of $risk_j$.

Example: The risk 'Open NTP' has an average amplification factor of 557.[4]

$risk_j(ASN_i)$ the number of all distinct IP addresses which are vulnerable to the jth risk within a specific ASN_i.

Similarly we can define the combined risk of a $country_i$ by:

$$risk_{country_i} = \sum_{j=1}^{n} risk_j(country_i) * AF(risk_j)$$

$risk_j$ is the number of all distinct IP addresses which are vulnerable to the jth risk within a specific $country_i$. The geolocation of IP addresses to countries needs to happen in a consistent way which also respects historic IP to country assignments (for example via maxmind geoip[5])

Why was the rank() omitted in this variant?

The implementers considered ranking countries / ASNs in a table and thereby getting an implicit rank (by row position in the table)[6].

This lead to an interesting interpretation of metrics version 2.1: it's trivial to estimate a *global DDoS potential* for the Internet, for ASNs or for countries: the idea is to assume an average of some fixed MBit/sec upstream for each vulnerable device and multiply the number of vulnerable devices of a country or ASN (as defined above) with the that MBit/sec upstream[7]. Thus we arrive

[2] `http://www.cybergreen.net/img/medialibrary/CyberGreen%20Metrics%20v.2.pdf`

[3] See the discussion in github issue 99

[4] Source: US-CERT `https://www.us-cert.gov/ncas/alerts/TA14-017A`

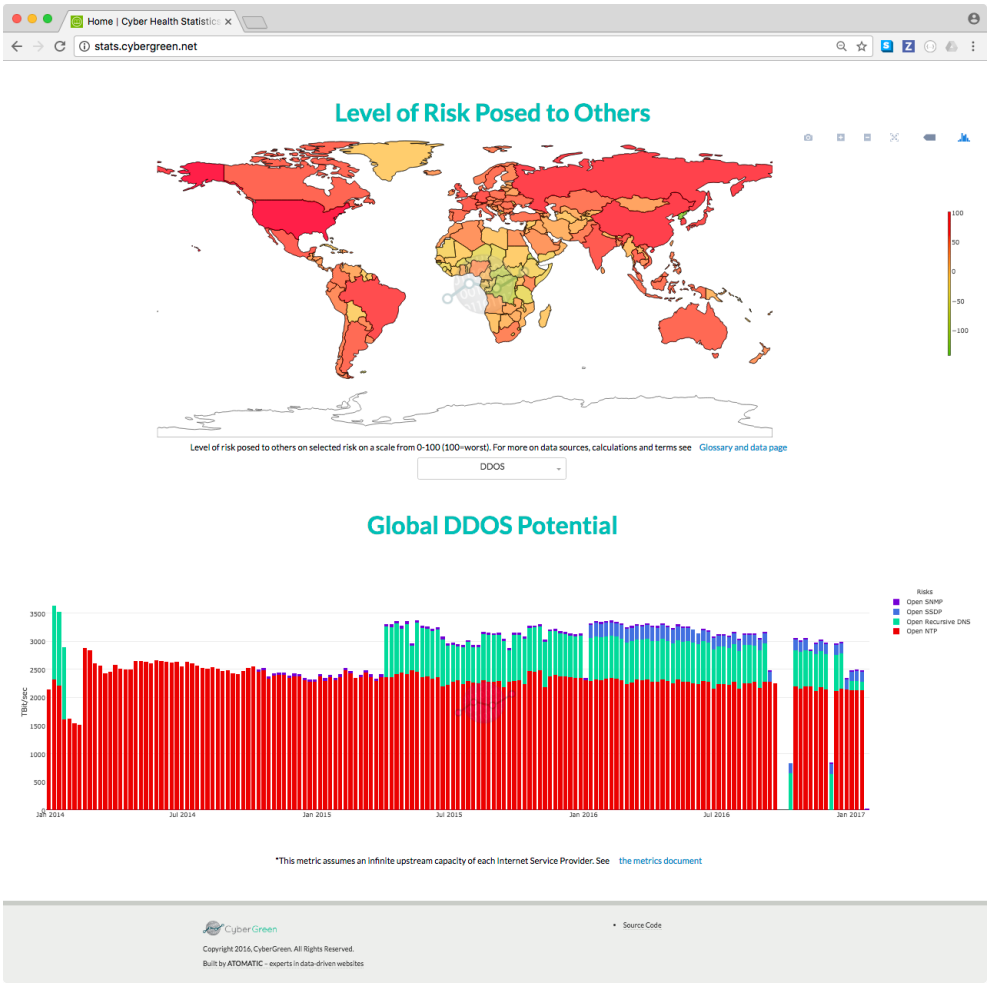
[5] `https://maxmind.com`

[6] `https://github.com/cybergreen-net/pm/issues/97`

[7] This estimate was done based on MLAB's data. Our current number of 4.91 MBit/sec is simply the average upstream speed of every client.

at the concept of 'DDoS firepower' (currently in the single to two digit TBit/sec range per country - for our data).

This application and slight variation of the CyberGreen metric version 2.1 raised eyebrows in the insurance industry since it gave the industry quantifiable numbers for estimating a maximum DDoS impact.

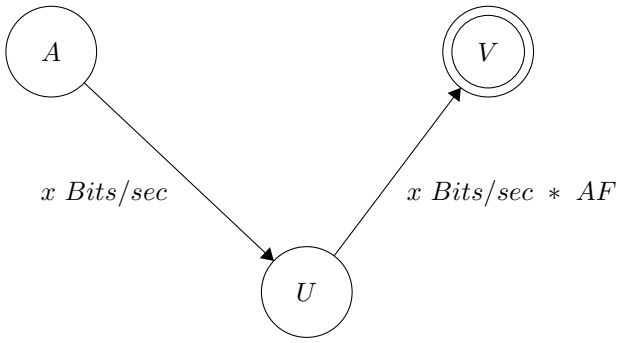


global DDOS potential visualization: This visualization is based on metrics 2.1. Note the missing data for the risk 'Open recursive DNS' and 'SSDP' during certain times.

A flaw in the naive assumption of metrics version 2.1

The simple idea of multiplying the risk by some constant AF is tempting but it ignores an important constraint of real world networks: *upstream of an amplifier is limited*.

Let's look at a model of an amplification attack:



Attacker **A** sends x Bits/sec via a **U** (UDP amplifier), which then amplifies the data to the victim **V** by an AF. However, U's upstream is capped.

Enter metrics version 2.1.5

As a proposed intermediate step between metrics 2.1 and metrics 2.2 ("normalization") we propose to fix the capped upstream issue.

$$DDOS_{c_i}[\mathbf{MBit/sec}] = \sum_{j=1}^{n} risk_j(c_i) * \min\left(US(c_i); AF(risk_j) * DS(c_i)\right)$$

Where $US(c_i)$ is the averaged upstream for a country c_i and $DS(c_i)$ is the average downstream. It might make sense to think more about means than average. US and DS are seen as from the perspective of the device.

We can further refine this metric since there are a couple of datasets which give us an upstream data for individual IP addresses, netblocks, ASNs or countries. MLAB [8] is such an example. It is about to release an API which will make it possible to refine this estimate on a per ASN level, from empirical sources[9].

With the new MLAB data we will be able to make a weighted sum over all individual IPs of amplifiers and their respective (averaged) upstream. Note that this then basically becomes the sum of the average upstream speeds of the vulnerable IP addresses.

Summary of 2.1.x

In any case, metrics version 2.1 does not work (even with a pure rank()) since it assumes an infinite upstream at every ISP. This is not realistic. The real *risk to others* posed by UDP based DDoS amplifiers must be measured differently. Metrics 2.1.5 is a first step at fixing the capped upstream issue. A better and

[8] `measurementlab.net`
[9] `http://data-api.measurementlab.net/`

more detailed version will sum up a country's or ASNs' vulnerable devices' upstream limits:

$$CG_i = rank\left(\sum_{i=1}^{N}\sum_{j=1}^{K} min(US(d_j); AF(risk_i) * DS(d_j))\right)$$

Where d_j is $device_j$. US is the averaged upstream speed of d_j. Similar for DS as downstream. We assume K vulnerable devices in a country/ASN.

Let's call this **metrics version 2.1.7**. We can do this with MLAB's[10] or ookla's data (as previously mentioned above).

Normalization

While the CyberGreen metrics v2.1.x explored the directions of how much of a *risk to others* a country or ASN could pose (if participating in a UDP based DDoS amplification attack), metrics v2.1.x does not allow networks owners or policy makers to identify how clean or dirty an ASN or country is when compared to others *in a relative way*.

If policy makers were to approach a government with the request to clean up their part of the Internet, they will need some relative score. Additionally, during multiple user interviews, CyberGreen repeatedly heard the wish to compare one's own ASN or country against other ASNs or countries in a way which would show how the clean up rate / progress was going.

For answering these questions, we have two paths:

1. show clean up rates (first or second derivatives), or
2. normalization

While the authors believe that the first approach might be promising as well, we will for now only concentrate on the second approach.

Once normalization enters the field, the immediate question is: how to normalize? By which factors? The subsequent sections will explore possible values to normalize by and will list pros and cons of each approach.

By socio-economic indicators

By GDP

- ⊕ Intuitive,
- ⊕ We have good and solid reference data available (worldbank.org, ...)
- ⊖ Joe St. Sauver's argument[11]: we want to show 'pollution' (in absolute numbers). Not relative pollution. Relative pollution might lead countries with a high GDP to believe they don't have to do anything.

[10] `data-api.measurementlab.net`
[11] XXX insert reference

⊖ Countries posing a bigger risk to others might get normalized out (appear good) because of a higher GDP.

⊖ The Internet is not really built for borders. Hence geolocation is always a bit incorrect.

⊖ It's not clear how and if GDP is somehow related to (UDP DDOS potential) 'pollution'

⊖ Assigning a fraction of the country's GDP to an ASN is a tricky undertaking, especially since ASNs are often trans-national.

By human population

All the very same arguments as stated in the GDP section apply here as well. Both on the pro as on the con side.

other socio economic indicators

Most of these have the same advantages and drawbacks as the GDP.

By Internet-technical indicators

By machine/service count

Another approach would be to normalize by number of actual (running, active) devices (servers, PCs, etc) in an ASN or country. Or by the number of services in an ASN/country (note: one device can run multiple Internet-facing services such as a HTTP web server).

⊕ Gives a precise 'vulnerable' versus 'in use' ratio.

⊖ Very hard to obtain precise data on the number of active devices or services: service discovery is non-trivial. Often the knowledge of particular hostnames is needed for HTTP or HTTPs SNI services. Where to get these hostnames from? IP addresses might not answer to a ping in the first place. And even if they do, it's not an indication of an 'active machine'. It might be just a ping answer of a honeypot. Other machines are active but might not answer to a ping. In summary, for precise numbers on the count of machines, we would need global and timely netflow data.

⊖ machine/service count fluctuates constantly for a variety of reasons

whole netblocks get announced/de-announced via BGP.

time of day in place scanned

changes in RTT

difference in protocol timeouts

The authors believe that while this normalization might make the most sense, it is currently very hard to obtain precise data.

By announced address space

This approach is similar to the machine/service count approach but approximates it a bit: we assume every announced IP address is announced (in BGP) because it shall run a service. Clearly this is not always the case. Sometimes, non-active netblocks are announced in BGP[12]. But over time, it will be representative - especially since the IPv4 space gets more and more crowded and every free netblock will eventually actually being used ('active').

- ⊕ Gives a good proportional 'vulnerable' versus 'in use' rate. Not as precise as the actual number of machines or services per ASN or country, but it is an approximation.
- ⊕ Easy to obtain precise data. Just have to fetch the announced netblocks per ASN from a live BGP table.
- ⊕ Easy to do backwards into history since historical BGP tables exist.
- ⊖ The numbers fluctuate regularly. Think: whole netblocks get announced/de-announced via BGP. Hence we will have 'jumpy' graphs.
- ⊖ BGP tables might differ over the globe. The data collection therefore varies with the vantage point of collection.

With this normalization method, we will have to do some more statistical analysis how ASNs tend to announce their assigned netblocks. Note the assumption of a correlation between number of announced IPs and number of devices. There might be different policies (not to announce or intentionally announce even if there are no services running in a netblock). What about intentional IP darkspace ('network telescopes'[13])?

By assigned address space

This normalization method is similar to the one above, however it does not reflect on how much of an assigned IP block is actually announced in BGP (and thus reachable on the Internet).

- ⊕ Same as announced address space, however this number does not fluctuate so much since assignments don't.
- ⊕ Easy to obtain precise data. Just have to fetch list of registered ASNs and the assignement sizes from the RIRs.
- ⊖ Not as precise as the number of announced addresses.
- ⊖ large legacy ASNs from the beginning times of the Internet have an unfair advantage: they were assigned huge blocks.

[12] For example to prevent spammers from hijacking this netblock

[13] `https://www.caida.org/projects/network_telescope/`

Proposed next metric version 2.2

The authors propose to use - *for this next iteration of CyberGreen metrics discussions* - to go with the number of assigned IP addresses or with the number of announced IP addresses and see how well these densities perform, especially how well they can serve in answering questions to the data.

Future work

The authors believe that over time CyberGreen will have multiple metrics in parallel. Depending on the question asked, we might need a variety of different metrics. One example was the question: 'what could be the maximum DDoS hitting my network in TB/s?'. This question is answerable with metric version 2.1 but not with 2.2.

Therefore, the next research steps may be:

1. collect more reference data for metrics
2. make sure that historic reference data is kept as part of the data sourcing process
3. experiment with new questions and metrics which could answer these questions
4. explore rates of change (1st and 2nd derivatives).

We want to hint at another approach which the authors feel would be very promising: the definition of a cost based metric: what are the costs for an attacker, what are the costs for a defender?

In particular one way to normalize vulnerable or infected counts is to price them according to an attackers cost to find them. In other words, if it costs X to scan a space, and Y vulnerable devices are found in that space then X/Y.

Acknowledgments

The authors would like to thank the CyberGreen Stats-WG especially Dan Geer and Manel Medina and Joe St. Sauver for their input and previous work.