Resilience Metrics

Valentin Zieglmeier
Advisor: Dr. Heiko Niedermayer
Seminar Future Internet SS2016
Chair of Network Architectures and Services
TUM Department of Informatics
Email: v.zieglmeier@tum.de

ABSTRACT

Computer networks have become an essential part of our world. To ensure their operational safety, measures need to be taken. Resilience defines how well a network can maintain operational safety in the face of various challenges to its operation. Metrics are needed for each aspect of resilience to enable quantification of the resilience of networks.
In this paper the different disciplines of resilience are named and explained. For each discipline the corresponding resilience metric is presented. To illustrate the need for resilience in practice, two case studies and possible solution strategies utilizing resilience metrics for each example are discussed.

Keywords

Resilience; Network security; Security measurement

1. INTRODUCTION AND MOTIVATION

Computer networks are ubiquitous today. Not only do we interact with them every day and rely on them in our personal life, we need them even more than we might sometimes think. Financial services can only be provided if we can rely on stable networks that are secure and safe. The military relies on networks to control troop movement, to plan attacks and to monitor drones. And many services we personally use are provided only through the internet.
An example for this are cloud services used to back up data and photos. Users of these services depend on the cloud service to work more reliable and to be more resilient than their personal computers. Another example is the Nextbit Robin, a smartphone that uploads unused apps and media from the smartphone to the cloud. When the user wants to access this uploaded content, it has to be downloaded first [17]. Lastly, services like Netflix aim to substitute the personal video collection with on-demand streaming. To watch a movie, the customer simply starts streaming it from the server [16].
Because it is important that these networks are reliable and work consistently, efforts are being made to ensure their resilience. To quantify the resilience of a network, we need to be able to measure it. In this paper a common definition of resilience is presented in section 2. Section 3 defines aspects of resilience and a corresponding metric for each. In section 4 case studies of real world examples are discussed. Finally, section 5 gives a summary and a conclusion.

2. DEFINITION: RESILIENCE

Resilience is no fixed concept. One definition can be found in [6, p. 6], [20, p. 1246] and [21, p. 12]:

> Resilience is the ability of the network to provide and maintain an acceptable level of service in the face of various faults and challenges to normal operation.

We assume that the network is under constant threat. As it is extremely important that we can rely on certain networks to function, the challenges that they face must not jeopardize their operations. Resilience is a measure to guarantee an acceptable level of service in the face of these challenges. Resilience metrics may be used to quantify how well a network can retain this level of service regarding different challenges.
As a prerequisite, common challenge categories need to be defined first. There are five basic categories of challenges for networks, listed below [6, p. 3].

- Environmental (e.g. node mobility),
- Malicious (attacks),
- Non-malicious (e.g. unusually high traffic load),
- Large scale disasters (e.g. a hurricane) and
- Lower level failures (e.g. path failures).

One strategy to achieve resilience is $D^2R^2 + DR$ (*Defend*, *Detect*, *Remediate*, *Recover* + *Diagnose*, *Refine*; see figure 1). It is at the core of the ResiliNets [20, p. 1253] and the ResumeNet projects [21, p. 24]. Both of these projects research resilience metrics and developed a framework to achieve resilience in networks.

This strategy is based on the idea that unforeseen events will always occur. After installing general defensive measures, these events have to be detected and the defensive measures strengthened to react to similar challenges more appropriately. An example for this might be failed links in a network. Remediation in this case would mean that traffic is rerouted. After the damage has been repaired or the challenge has been overcome, the recovery is initiated. In this stage the system returns to its normal state [20, p. 1254].
These measures are assumed to always have flaws, so they have to be constantly diagnosed and refined. In case the automatic measures were not enough to ward off the challenge, they may be developed further [20, pp. 1254-1255].

doi: 10.2313/NET-2016-09-1_02

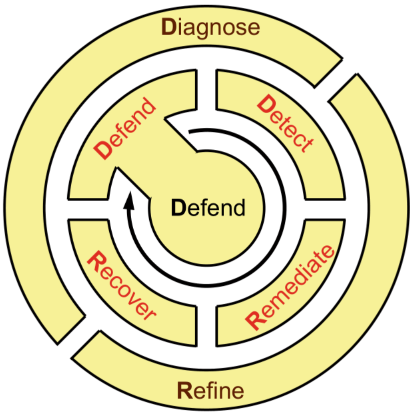

Figure 1: The $D^2R^2 + DR$ strategy of the ResiliNets project [20, p. 1253].

3. RESILIENCE METRICS

As we found in the last chapter, resilience includes tolerance of multiple different threat categories. Below we discuss the different disciplines of resilience and how each metric can be calculated or measured. The dimensions of resilience are adapted from [20, pp. 1247-1249] (see figure 2).

3.1 Challenge tolerance

Networks regularly face challenges. Their ability to tolerate them and to continue operating even when challenged defines one of the two main dimensions of their resilience.

3.1.1 Survivability

Survivability is defined as the "capability of a system to fulfill [sic] its mission, in a timely manner, in the presence of threats such as attacks or large-scale natural disasters" [20, p. 1247]. This refers to the general failure tolerance of a system. "Failure" refers to "correlated failures" [20, p. 1247] either because of an attack, or because large parts of the system fail.
As fault tolerance (see section 3.1.2) is defined as a subset of survivability, we can conclude that survivability is defined as a combination of failure tolerance and fault tolerance.
Survivability defines how well the system can maintain an acceptable level of service when parts of the system fail or are attacked. It does not include the ability of the system to prevent such attacks or failures.
The second part of survivability, fault tolerance, is discussed in the next section.

Metric. One definition of a method to assess survivability can be found in [10, pp. 8-9]:

> A system is survivable if it complies with its survivability specification. [...] A survivability specification is a six-tuple, {S, E, D, V, T, P}.

The six-tuple consists of the acceptable service specifications (S), all theoretical ways the system can degrade (E), all realistically reachable degradation states (D), an ordering of system specifications from S for certain degradation states from D from the perspective of the user (V), valid transitions between acceptable forms of service (T) and service probabilities for each member of S (P) [10, pp. 9-10].
By defining these different properties and analyzing them, the necessary survivability of a network can be assessed in consideration of the context.

An example might be a personal backup server. This server is connected to a RAID 1 with two hard drives that both contain the same data. Possible service states (S) might be:
s_1: Direct monitoring done at the server
s_2: Remote monitoring with email alerts for failures
s_3: No monitoring (e.g. when travelling)
Theoretical ways the system may degrade could be (reduced for simplicity):
E: {drive status(d) ↦ {good, one failed, both failed},
energy(e) ↦ {good, none}}
These are then combined into all possible combinations and realistic combinations are collected (D). In the example we use, all combinations are possible:
d_1: {d ↦ good, e ↦ good}
d_2: {d ↦ one failed, e ↦ good}
d_3: {d ↦ both failed, e ↦ good}
d_4: {d ↦ good, e ↦ none}
d_5: {d ↦ one failed, e ↦ none}
d_6: {d ↦ both failed, e ↦ none}
With this we can define priorities of service states for each degradation state (V), as seen exemplary in table 1.

Table 1: Example of V: Higher is better

	s_1	s_2	s_3
d_1	2	3	1
d_2	3	2	1
d_3	3	2	1
d_4	3	1	2
d_5	3	1	2
d_6	3	1	2

Continuing with T, possible transitions have to be defined. In our example, every state is accessible from every state. This can for example be marked in a graphic containing the service states, with arrows connecting them for every possible transition.
Lastly for P, probabilities of each service state are calculated. These are the percentages of time that the system will probably be in the respective state. In our case that might be:
$\Pr[s_1] = 0.5$; $\Pr[s_2] = 0.3$; $\Pr[s_3] = 0.2$
This concludes the exemplary survivability specification.

3.1.2 Fault tolerance

Fault tolerance is defined as a subset of survivability. It is "the ability of a system to tolerate faults such that service failures do not result" [20, p. 1247]. Contrary to failures, faults are "random uncorrelated failure[s] of components" [20, p. 1247].
This metric does not define if such faults can be prevented, but how well the system can handle them. The goal is to provide acceptable service to the users.

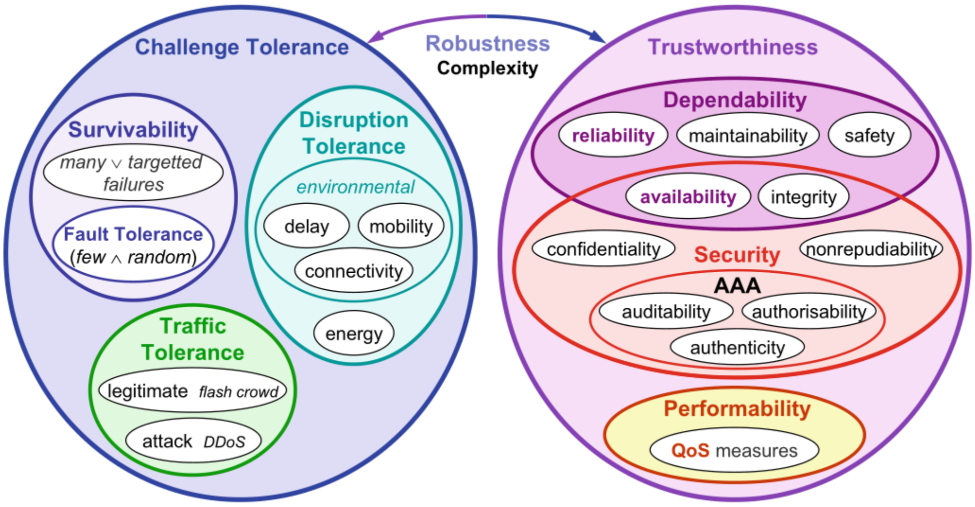

Figure 2: The different resilience disciplines [20, p. 1247]

Metric. As fault tolerance can be achieved with redundancy, a measurement for good fault tolerance might be the number of redundant instances available. Assessing how much is "good enough" is depending on the context. For military or financial applications this might be significantly higher than for private applications.

3.1.3 Disruption tolerance

Disruption tolerance is defined as the "ability of a system to tolerate disruptions in connectivity among its components" [20, p. 1248]. It consists of multiple possible disruptions, grouped as environmental and energy challenges. Environmental challenges can be delays, mobility and weak connectivity [20, p. 1248]. Node mobility refers to nodes that change their position, which influences connectivity and changes the wireless topology of the network. An example for this are mobile phones.

This metric is similar to survivability, but at a smaller scale. Disruption tolerance can refer to the system tolerating the failure of just a few connections, while survivability may refer to a power grid failing, resulting in the outage of a complete local network.

Metric. The disruption tolerance could be measured as the percentage of possible disruptions the system can tolerate or as the percentage of system parts that may be disrupted without the system failing. Which aspects are most important again depends on the context.

To handle different possible disruptions, redundancy of multiple types should be in place. This includes systems utilizing different technologies and hardware or being installed in multiple geographic locations.

3.1.4 Traffic tolerance

The last aspect of challenge tolerance, traffic tolerance, is defined as "the ability of a system to tolerate unpredictable offered load without a significant drop in carried load (including congestion collapse), as well as to isolate the effects from cross traffic, other flows, and other nodes" [20, p. 1248]. Traffic tolerance is a metric that can be observed when it is too low. If a higher-than-usual amount of users try to connect to a server at the same time and that server cannot be reached any more as a consequence, the traffic tolerance is inadequate. This is called an "unusual but legitimate traffic load" [20, 1249]. An example for that might be many students trying to register for a course at the same time. This problem can be especially severe if the network fails just when it is needed the most.

Metric. Traffic tolerance can be measured by testing how much traffic the system can handle while still offering acceptable service to legitimate users. Different tests can include traffic from an identical IP address as well as from different IP addresses (similar to a simple distributed denial-of-service attack).

To test various more advanced tolerance strategies the requests can be modeled after realistic and unrealistic client behavior. Possible patterns from legitimate flash events [7, pp. 3-5] or from distributed denial-of-service (referred to as DDoS) attacks [7, pp. 5-7] can be taken into consideration.

3.2 Trustworthiness

The second dimension of resilience is trustworthiness. This refers to the predictability of the system from the perspective of the consumers of its services.

3.2.1 Dependability

The first aspect of trustworthiness is defined as the dependability of the system, which "quantifies the reliance that can be placed on the service delivered by a system" [20, p. 1248]. The main parts of this are availability and reliability.

Availability can be defined as "readiness for correct service" [3, p. 6]. It is the percentage of time the system is available in contrast to being unavailable. This metric is important for servers that are accessed regularly and whose availability is important to the business of a company, like shopping sites. More than 80 % of mobile users will abandon a site if it loads longer than 20 seconds or not at all [8].

Reliability by contrast is "continuity of correct service" [3, p. 6]. This can be understood as how long the service is available without interruption. This metric is important e.g. for video streaming or voice over IP services, where a continuous connection is necessary to maintain an acceptable level of service for users.

The remaining parts of dependability are maintainability, i.e. the "ability to undergo modifications, [sic] and repairs" [3, p. 6], safety, i.e. the "absence of catastrophic consequences on the user(s) and the environment" [3, p. 6] (protection *from* the system in contrast to security, protection *of* the system) and integrity, i.e. the "absence of improper system alterations" [3, p. 6] ("improper" means "unauthorized" here [3, p. 6]).

Metric. The availability of a system is defined as

A = MTTF/MTBF [20, p. 1248].

The mean time between failures (MTBF) is defined as

MTBF = MTTF + MTTR [20, p. 1248].

The mean time to failure (MTTF) is the mean of continuous service uptime periods. The mean time to recovery (MTTR) is the mean of continuous service downtime periods. This means the availability is equivalent with the percentage of time that the system was available.

The reliability is depending on the period of time that the system should work continuously without failing. After defining this period, the reliability can be calculated as

$R(t)$ = Pr[no failure in [0,t]] [20, p. 1248].

It is the probability that the system may fail during the specified period of time t.

Maintainability, safety and integrity are qualitative and binary properties, meaning that they are either true or false and this depends on the requirements. They need to be assessed and prioritized in consideration of the context.

3.2.2 Security

The second aspect of trustworthiness, security, is defined as "the property of a system, and the measures taken such that it protects itself from unauthorised access or change, subject to policy" [20, p. 1249]. Availability and integrity, which were defined as parts of dependability, are shared with security (see figure 2). Additional properties are authenticity, authorizability, auditability, confidentiality, and nonrepudiability [20, p. 1249].

In contrast to the other disciplines, security is not aimed at keeping the network running at any cost. The target is to prevent malicious access and modification. This can mean shutting down or cutting off parts of the network that might be compromised.

Metric. Metrics for availability and integrity are discussed in section 3.2.1. Authenticity, authorizability, auditability, confidentiality and nonrepudiability have to be assessed in a security audit. If that is possible, auditability is given and the other properties can be analyzed. These too are qualitative and binary properties, so this assessment is based on the requirements. If a security audit is not possible, they cannot be assessed. In general, it is difficult to measure and compare these properties.

3.2.3 Performability

The last aspect of trustworthiness, performability, is defined as the "property of a system such that it delivers performance required by the service specification" [20, p. 1249]. Similar to traffic tolerance, inadequate performability can be experienced directly by the users and affect their behavior. Ten seconds of loading time can make almost 50 % of mobile users abandon the page completely [8].

Metric. Performability is another requirement that is almost completely depending on the context. How much delay or throughput is necessary for acceptable service has to be decided on.

3.3 The connection between challenge tolerance and trustworthiness

To describe how the two dimensions challenge tolerance and trustworthiness are interconnected, the following aspects are named.

3.3.1 Robustness

The robustness of a system is defined as "the trustworthiness [...] of a system in the face of challenges that change its behaviour" [20, p. 1249]. This is exactly the interconnection of challenge tolerance and trustworthiness. Robustness therefore quantifies how trustworthy a system remains when challenged. It should ideally not change.

Other definitions of robustness are closer to survivability, like that "internet communication must continue despite loss of networks or gateways/routers" [23, p. 6].

Metric. Some metrics that robustness is based on are often not clearly quantifiable, like security or dependability, making the quantification of robustness also inexact. In general, a better challenge tolerance of a system means that its robustness is better as well.

3.3.2 Complexity

Complexity is named as the result of resilience mechanisms. If more of these mechanisms are deployed, the system gets more complex which "may result in greater network vulnerability" [20, p. 1249].

This means that work on resilience may also reduce the resilience regarding other aspects of the network.

Metric. Complexity increases "may be related to maximization of the information shared (transferred) within the system" [18, p. 4]. This does not mean that more shared information leads to higher complexity, but a lower amount of information that is shared may indicate a lower complexity of the network.

4. CASE STUDIES

Below we analyze two real world cases where resilience was a factor. This can help to understand the importance of different aspects of resilience.

First, we analyze Hurricane Katrina to see what impact it

had on network infrastructure and to gain a better understanding of the importance of survivability and dependability.
Second, we analyze the 2007 cyber-attacks on Estonia directed from Russia. That example illustrates the need for good traffic tolerance, especially regarding networks that have an increased risk to be attacked.

4.1 Case study 1: Hurricane Katrina

Hurricane Katrina is one of the best-known large-scale natural disasters to hit the United States. This hurricane was "the costliest and one of the five deadliest hurricanes to ever strike the United States" [9, p. 1]. Between 23rd and 30th of August 2005 at least 1.245 people died as a consequence. The property damage is estimated to have been around $ 108 billion [9, p. 1].
Even though such large-scale natural disasters are not common, it is extremely important that people can still communicate with each other and learn about safety measures or whether they are affected in similar cases of emergency. Additionally, the regular network service in other states or countries should neither be affected by such disasters.

4.1.1 *Impact of Hurricane Katrina on network infrastructure*

A report published by Renesys found that over 35 % of networks in Mississippi, over 10 % in Louisiana and over 5 % in Alabama among others were outaged during Hurricane Katrina [4, p. 2] (see figure 3).

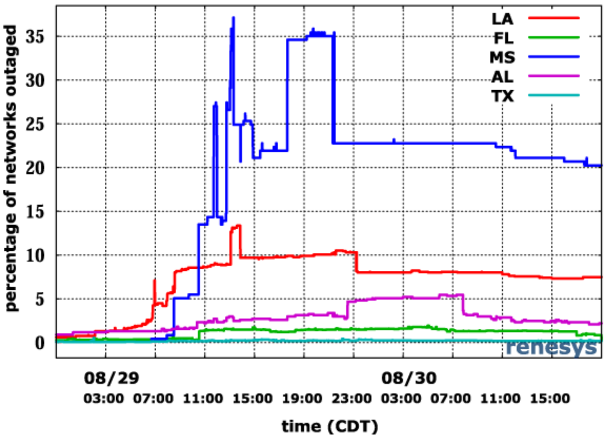

Figure 3: Percentage of globally routed networks outaged by Hurricane Katrina, as measured by Renesys [4, p. 3].

While the early outages were quickly fixed and the network operation recovered, the networks experienced extended outages in the two most affected regions, Mississippi and Louisiana, over the entire 10-day period (see figure 4).

4.1.2 *Survivability through path diversification*

There is no data about how these outages affected users of the network, but it can be assumed that the significant effect on the network infrastructure of the region affected them as well. Without necessary survivability measures, the adequate dependability for the users could not have been maintained. It is very important to be able to keep the network stable even if significant parts of it fail.

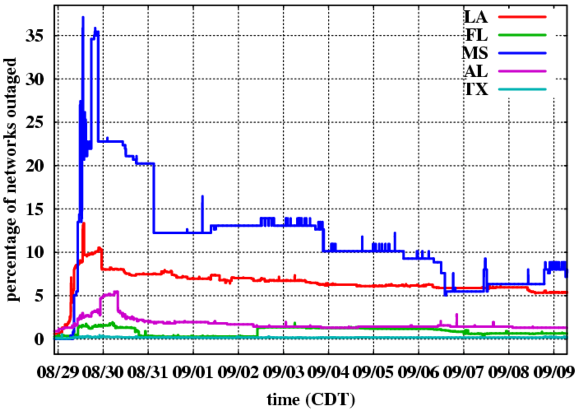

Figure 4: Percentage of globally routed networks outaged by Hurricane Katrina over the entire 10-day period since 29th of August, as measured by Renesys [4, p. 7].

To prevent such disasters from affecting the whole network, redundant systems should be in place. These systems should be located away from each other with enough distance between them that only one of them can be affected at once. Also the network should be able to dynamically reroute requests in case a part of it has failed. This approach is referred to as path diversification, as multiple possible paths for connections in the network are introduced by it. It is already common practice, but cannot be easily controlled and measured.

There has been research in the field of path diversification with the goal of achieving a high flow reliability [19]. Diversity of two paths P_a and P_b is defined as
$D(P_b, P_a) = 1 - \frac{|P_b \cap P_a|}{|P_a|}$ [19, p. 345].
For two completely disjoint paths this formula equates to 1, for equal paths to 0. (The endpoints of the paths are excluded as they are always identical.)
It is important to note that neither node nor link diversity alone is enough, as proven by [19, p. 345]. To be a valid fallback, the paths have to be disjoint regarding both links and nodes.
With this definition the path diversity of a network can be assessed and ensured. To make sure that a failing path cannot cause a network outage, the diversity of at least two paths has to be 1, meaning that they share no node or link. That means in case a node or link fails there is always at least one alternative path.
This can be further improved by taking past node or link failures into account and ensuring that fallback paths in different geographic locations or utilizing completely different technologies (e.g. optical and satellite) exist.

4.1.3 *Possible rerouting strategies*

To ensure that packages can be rerouted, a possible scenario is the following. Each routing table gets additional alternative routes for each node in the network. Should a link fail the node can propagate that network change and reroute packages accordingly [14, pp. 2-4].

An alternative rerouting strategy is the proactive failure insensitive routing, proposed by [12]. The idea is to reroute the packages locally when a link fails. Instead of explicitly notifying other nodes of the failure, they can infer it from the packets they receive. If a packet is received on an unusual interface, it was probably rerouted. This means no changes to the link state propagation mechanism are necessary in this approach [12, p. 2].

4.2 Case study 2: Estonia under cyber-attack

In April 2007 Estonia provoked Russia by removing a war statue honoring soviet war dead from World War II from the city center of Tallinn. This led to protests from Russians in Estonia and Russia and violent riots in Tallinn. At the same time as the protests began, DDoS attacks started hitting Estonian servers [13]. After only affecting the website of the foreign minister of Estonia, they quickly spread. The main targets were the website of the Estonian police, the Ministry of Finance and other government websites [2]. Origin of the attack were Russian IP addresses which lead the Estonian government to claim involvement of the Russian government [13, p. 1]. This claim was questioned by security researcher Mikko Hyppönen of F-Secure in the *Helsingin Sanomat* [1]. While the load was not extraordinarily high, it hit Estonia badly because the country is small and its networks are not prepared for such load [13, p. 1].

4.2.1 Impact of the attacks on the network

The attacks were no steady stream and not of equal length, they were distributed over many days and different attack strengths (see table 2).

Table 2: Attack lengths and bandwidths [2]

Attacks	Length
17	< 1 minute
78	1 min - 1 hour
16	1-5 hours
8	5-9 hours
7	≥ 10 hours

Attacks	Bandwidth
42	< 10 Mbps
52	10-30 Mbps
22	30-70 Mbps
12	70-95 Mbps

The impact of the attacks was enormous. Analyses by Mikko Hyppönen show that the homepage of the Estonian government was hardly accessible if at all (see figure 5).

Estonia tried to counter these attacks, but failed to do so. The countermeasures were not effective, so they had to cut their connection to the rest of the internet to restore services within Estonia for their population, many of whom relied on these services for their everyday life [13, p. 3].
This measure was the last resort and it helped. But it resulted in degraded service for the Estonian people. Internet access was effectively cut, meaning the only websites they could visit were Estonian. Accessing ATMs from other countries was now also almost impossible [13, p. 3].
Should the attacks have been launched from inside Estonia, this measure would have been of little use. In any case, if the attacks did not stop after some time, it would not have resolved the issue.

4.2.2 Traffic tolerance through resource accounting

DDoS attacks aim to prevent legitimate users from accessing a service. Either the users are directly attacked, or the server they try to access is targeted [24, p. 1], [22].

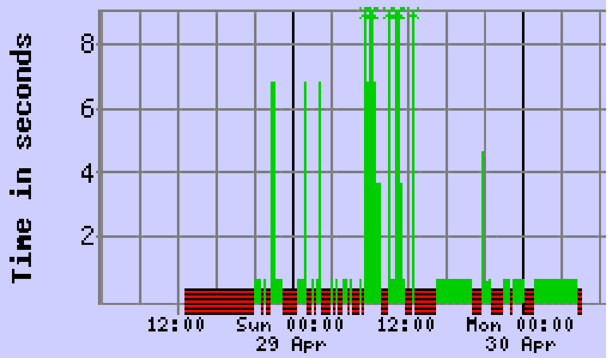

Figure 5: Availability statistics from Netcraft for the homepage of the Estonian government (cropped) [5]. Green (light) are successful connections, red (dark and horizontally striped) are failures.

These attacks work because servers discard requests when they cannot handle them anymore. If a server follows this strategy, some legitimate requests will be discarded and some illegitimate requests handled which results in degraded service for legitimate users. One possible strategy to prevent this denial-of-service is resource accounting [15, p. 10]. In this scenario the requests are observed and discarded based on their properties and predefined rules. This aims to filter out more illegitimate requests and provide service to actual users.

4.2.3 A possible resource accounting strategy

Two properties seem to be very effective to detect illegitimate requests: The per-client request rate and cluster overlap before and during the event [7, p. 7].
The per-client request rate during DoS attacks was found to differentiate compared to legitimate flash events. During flash events it will slow down as the server responds slower to requests. In contrast to that the request rate will stay constant during DDoS attacks [7, p. 8].
To further refine which requests are discarded, the server can periodically cluster the legitimate requests during normal load. A possible clustering strategy is the network-aware clustering, for example by using the method proposed by Krishnamurthy and Wang [11]. When the server experiences unusually high load it can compare the requests to the clusters from past requests and discard unknown clients first. They are more likely to be attackers [7, pp. 7-8].

5. CONCLUSION

Networks have become the backbone of our society, and we depend on them working consistently and continuously. Challenges to networks occur constantly and come from various sources. To withstand these challenges while maintaining acceptable levels of service, networks need to be resilient. To ensure the resilience of a network, agreed on metrics that quantify the different aspects of resilience are necessary. These metrics can be used to assess the resilience of a network as well as to compare different approaches to achieve resilience and to understand the consequences of protective measures.

In this paper a common definition of resilience was presented. A metric was found for every aspect of resilience. Two aspects have been analyzed more extensively in two case studies and possible solution strategies for both cases have been presented.

It is possible to achieve basic resilience in a network based on the work in this paper. The presented metrics provide a basic understanding of resilience and its aspects. The two discussed case studies and the presented solution strategies offer profound information regarding similar scenarios.

The remaining aspects of resilience may be studied in more detail and the corresponding metrics can be refined. To find out which countermeasures are most effective, some resilience disciplines need to be researched more extensively and experimented with.

6. REFERENCES

[1] N. Anderson. Massive ddos attacks target estonia; russia accused. http://bit.ly/221ZnDj, 2007. Accessed: 2016-04-03.

[2] ARBOR. Estonian ddos attacks - a summary to date. http://bit.ly/1Rturu3, 2007. Accessed: 2016-04-01.

[3] A. Avižienis, J.-C. Laprie, B. Randell, and C. Landwehr. Basic concepts and taxonomy of dependable and secure computing. *Dependable and Secure Computing, IEEE Transactions on*, 1(1):11–33, 2004.

[4] J. Cowie, A. Popescu, and T. Underwood. Impact of hurricane katrina on internet infrastructure. *Report, Renesys*, 2005.

[5] M. Hypponen. Update on the estonian ddos attacks. http://bit.ly/1T2Zkbk, 2007. Accessed: 2016-04-01.

[6] A. Jabbar. A framework to quantify network resilience and survivability. 2010.

[7] J. Jung, B. Krishnamurthy, and M. Rabinovich. Flash crowds and denial of service attacks: Characterization and implications for cdns and web sites. In *Proceedings of the 11th international conference on World Wide Web*, pages 293–304. ACM, 2002.

[8] KISSmetrics. How loading time affects your bottom line. https://blog.kissmetrics.com/loading-time/, 2011. Accessed: 2016-03-31.

[9] R. D. Knabb, J. R. Rhome, and D. P. Brown. *Tropical cyclone report: Hurricane katrina, 23-30 august 2005.* National Hurricane Center, 2005.

[10] J. C. Knight, E. A. Strunk, and K. J. Sullivan. Towards a rigorous definition of information system survivability. In *DARPA Information Survivability Conference and Exposition, 2003. Proceedings*, volume 1, pages 78–89. IEEE, 2003.

[11] B. Krishnamurthy and J. Wang. On network-aware clustering of web clients. 2000.

[12] S. Lee, Y. Yu, S. Nelakuditi, Z.-L. Zhang, and C.-N. Chuah. Proactive vs reactive approaches to failure resilient routing. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1. IEEE, 2004.

[13] M. Lesk. The new front line: Estonia under cyberassault. *Security & Privacy, IEEE*, 5(4):76–79, 2007.

[14] M. Menth and R. Martin. Network resilience through multi-topology routing. In *The 5th International Workshop on Design of Reliable Communication Networks*, pages 271–277, 2005.

[15] J. Mirkovic and P. Reiher. A taxonomy of ddos attack and ddos defense mechanisms. *ACM SIGCOMM Computer Communication Review*, 34(2):39–53, 2004.

[16] Netflix. How does netflix work? https://help.netflix.com/en/node/412, 2016. Accessed: 2016-04-01.

[17] Nextbit. Meet robin. https://www.nextbit.com/pages/meet-robin, 2016. Accessed: 2016-04-01.

[18] M. Prokopenko, F. Boschetti, and A. J. Ryan. An information-theoretic primer on complexity, self-organisation and emergence. *Advances in Complex Systems*, 2007.

[19] J. P. Rohrer, A. Jabbar, and J. P. Sterbenz. Path diversification: A multipath resilience mechanism. In *Proceedings of the IEEE 7th international workshop on the Design of Reliable Communication Networks (DRCN)*, pages 343–351, 2009.

[20] J. P. Sterbenz, D. Hutchison, E. K. Çetinkaya, A. Jabbar, J. P. Rohrer, M. Schöller, and P. Smith. Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines. *Computer Networks*, 54(8):1245–1265, 2010.

[21] P. Trimintzios. Measurement frameworks and metrics for resilient networks and services: Technical report, european network and information security agency (enisa). Technical report, Tech. Rep., February, 2011.

[22] US-CERT. Understanding denial-of-service attacks. https://www.us-cert.gov/ncas/tips/ST04-015, 2009. Accessed: 2016-04-02.

[23] W. Willinger and J. Doyle. Robustness and the internet: Design and evolution. *Robust-Design: A Repertoire of Biological, Ecological, and Engineering Case Studies*, pages 231–272, 2002.

[24] S. T. Zargar, J. Joshi, and D. Tipper. A survey of defense mechanisms against distributed denial of service (ddos) flooding attacks. *Communications Surveys & Tutorials, IEEE*, 15(4):2046–2069, 2013.