# CHAPTER 14

# The Notion of Security Risk

## 14.1 Risk As Combination of Probability and Severity

There is a new draft of ISO/IEC Guide 120, for the inclusion of cybersecurity considerations in international standards [4]. It considers security risk, as it must and as we must. However, the IEC notion of "risk" is defined in the International Electrotechnical Vocabulary [2] in the manner usual in safety as "*combination of probability.....and severity....*".

This notion is misplaced in the context of security, because the notion of probability is misplaced. Consider:

Suppose system software $S$ incorporates a *vulnerability*. A vulnerability is a mechanism $V$ allowing $S$ to enter a state *St-bad* from which $S$ causes the system to behave in a way which results in harm (let us say inevitably, for the purposes of this argument). There are two cases to consider:

1. *St-bad* might be simply provoked through giving input I to S
2. *St-bad* might be induced through internal manipulation (through intrusion or malware)

We can consider these two cases as both examples of case (2): given case (1), define *St-bad* as the the state of S when presented with input I. So we consider here just case (2).

When considering safety, we can ask how often a running S enters *St-bad* "in the

wild". We can expect that this occurs with a certain frequency, and this frequency yields the "probability" of harm (which inevitably follows *St-bad*). We may well not know what *St-bad* may be (else there would likely be a due-diligence requirement to inhibit S from entering this state).

## 14.2 The Notion of Security Risk

Now consider security. Let me not talk about the "probability", but about the *chances of harm*.

**Level 1** . V exists, but no one knows of it. Chances of harm through St-bad are the "in the wild" chances.

**Level 2** . V exists, and someone knows of it. However, they don't know what characterises St-bad and therefore what will activate V. Chances of harm through St-bad remain the same as in Level 1, the "in the wild" chances.

**Level 3** . V exists, someone knows of it, and they know what St-bad is. But they don't know how to induce S to enter St-bad. Define an "exploit" as a means of inducing a running S to enter St-bad through use of V. At this level, then, no exploit is known. Chances of harm through St-bad remain the same as in the previous two levels. However, the possibility that this will change depends inter alia on the difficulty of devising an exploit for V, and the motivation and capabilities of those who try to devise an exploit for V. If devising an exploit is easy, this situation and the chances of harm will change quickly.

**Level 4** V exists, and there exists an exploit E. The chances of harm depend then on social factors: someone who is sufficiently motivated to insert E into S, and has the mechanisms to try to do so. These mechanisms include access to S (which is often socially dependent; so-called "social engineering" techniques can yield access where it is unauthorised). I proposed it is inappropriate in engineering to speak of the "probability" of there being someone sufficiently motivated to insert E into S and sufficiently skilled to do so and who actually goes ahead to do it. However, at Level 4, the chances of S being compromised through E and its behaviour thereby resulting in harm is enormously higher than in the previous three levels.

**Level 5** V exists, there exists an exploit E, and this exploit E is inserted into S. The chances of harm resulting are then solely dependent upon the will and ability of someone to activate E. Again, it is inappropriate to speak in engineering terms of a "probability" here: what is the "probability" that an exploiter just had an argument with a lover and wants to wreak damage somewhere as a displacement activity? This personal state cannot be estimated using any concepts of engineering.

These levels represent different levels in exploiting V to lead to harm. I suggest that any exploit of V to lead to harm passes through these five levels, and that in at least two of those levels the notion of "probability" is inappropriate in engineering terms.

My suggestion follows of how one can define "security risk" appropriately.

## 14.3 Cybersecurity Phenomena in Safety-Related Systems

Ingo Rolle has pointed out that there are areas of electrotechnical engineering which might have safety-relevant cybersecurity vulnerabilities but whose safety is currently assured by well-tested traditional techniques [8]. For example, electrical safety is assured through overload- and residual-current-protection using digitally-based protection devices. Such devices could be deliberately compromised, vitiating the protection they offer.

ISO/IEC Guide 51 explains how safety is to be incorporated into standards for technical domains in which questions of safety arise [3].

The steps in Guide 51 are:

**Hazard Identification** A hazard is a "potential source of harm" (IEC 61508-4:2010), where harm is "physical injury or damage to the health of people or damage to property or the environment" (ibid.) Hazards are normally thought of as situations which are precursors to harm. Situations are often states, but can also be events.

**Risk Analysis** The severity of a hazard is assessed, that is, the worst case of harm which can arise through a hazard. Then the likelihood of the hazard occurring, along with the the likelihood that the severity will actually result from the occurrence of the hazard, is estimated. (What is actually wanted is the

expectation of loss, as it is known in statistics. It is often hard or impossible to distribute the loss across the range of possible outcomes, and so the worst case is taken in engineering as a conservative estimate of outcome.)

**Risk Assessment** The calculated risk is evaluated against the acceptable risk, and if it is higher, then.....

**Risk Reduction** Measures must be introduced into the system to reduce the risk to an acceptable level. Risk reduction measures in IEC 61508 and related standards involve

**Redesign** To avoid as much of the risk as possible, so-called safety functions are introduced. These are functions introduced explicitly to mitigate the hazard and/or its effects.

This is very different from the way in which, say, electrical safety is assured through venerable standards (Germany's standard for electrical safety of building-distribution systems, the series known as DIN VDE 0100, is 130 years old. It has been continually revised, of course). Very often, a basic protection is elaborated, along with secondary protection in case the basic protection fails. When devices intended to provide this protection are cybersecurity-vulnerable, a different approach, closer to that of Guide 51, may be required.

## 14.4 Malicious Intrusion Can Change Safety Parameters

Guide 51 is primarily about safety. There are ways in which malware or unauthorised intrusion into a system can affect its safety properties. The following observations are adapted from [7]. Malicious intrusion or malware can

**induce hazards** For example, consider a building electrical installation with a smart meter and Type B residual-current protection. Type B protection is dependent on programmable electronics; the programs are called firmware. High-end large-building protection devices are nowadays not necessarily isolated stand-alone kit, as a house Type B residual-current protection device still is. They can be reprogrammable in the field. Malicious interference with the smart meter could enable a manipulation of the firmware of the Type B device to render it malfunctional.

**elevate the frequency of hazards** The acceptability of the risk of a specific hazard is based on an estimate of how frequently it occurs. Such estimates are often based on experience. For example, let us say it is expected on the basis of experience that a specific high-end Type B residual-current protection device malfunctions once a year. The safety analysis takes that frequency into account. But an intruder coming in through a smart meter could reprogram the Type B device as above so that it malfunctions. This malfunction could be detected, and the device reset or replaced. But then, the next day, the intruder enters and causes the same malfunction with the new/reset device. The frequency of the hazard has been elevated to once a day, whereas the acceptable risk derived through the safety analysis was based on the frequency estimate of once a year.

**inhibit the proper operation of safety functions** The Type B device performs a continuity function, namely allowing electricity to flow when there is no detected waveform anomaly indicating residual-current flow, as well as a safety function, namely interrupting the flow when such a waveform is detected. The high-end reprogrammable Type B device could be manipulated so that it allows flow continuously, whatever the conditions, and fails to perform its safety function.

Cybersecurity measures may be necessary to prevent any of the above phenomena occurring. Thus cybersecurity is an important component in assuring functional safety.

## 14.5 Adapting the Concepts of Guide 51 to Cybersecurity

Suppose that a safety analysis along the lines of Guide 51 has been carried out; that hazards have been identified, risk assessed and risk reduction measures undertaken where necessary. The following security analysis builds on this information.

Define a *safety-related vulnerability* to be a means by which a hazard $H$ can be causally induced through malicious action (malaction). Malaction can be external, as with a DoS attack, exploiting service mechanisms which the system S offers to its environment, or it can be internal, involving modification of the system S itself, through insertion of new software code or modification of hardware. It can also be induced, activated through direct action of a human being, or automatic, executing autonomously when a certain system state is attained. (Any malaction

which combines induced and automatic components I classify as induced.)

The sr-vulnerability is said to be associated with the hazard $H$. Intuitively, a concrete safety-related security risk arises for system $S$ if a sr-vulnerability is known, an exploit exists, and there is a vector, namely a means of introducing the exploit into system $S$. If an exploit is introduced to $S$ via a vector, I say the exploit is installed in $S$. When an exploit installed in $S$ executes, I say the vulnerability is activated in $S$. I correspondingly define five levels of sr-vulnerability, which I shall call *srv-level*.

**SRV-Level 1** The sr-vulnerability is unknown to anybody. When sr-vulnerability is unknown , ipso facto it cannot be deliberately exploited. When a sr-vulnerability is unknown and cannot be exploited, a security risk arises through it becoming known and exploited, thereby raising the srv-level (see below). There is thus no intuitive security risk attached to srv-level 1.

**SRV-Level 2** The sr-vulnerability is known, but no exploit is known. When there is no possibility of exploiting an sr-vulnerability, then there is (as yet) no intuitive security risk. There is thus not yet an intuitively direct security risk attached to srv-level 2, but only a potential.

**SRV-Level 3** The sr-vulnerability is known and an exploit exists. There can be two levels of difficulty associated with an exploit. The first is that an exploit can be simple and straightforward, or it can be involved and ingenious, as well as everything in between. The difficulty of its inception is, however, secondary to its existence. The second level of difficulty is directly relevant to intuitive security risk. An exploit can be easy to induce in S, or difficult. Malware can immediately induce unsafe behaviour of a S, or it may need to wait for a rare system state which it can exploit to induce or increase the severity of a specific hazard. It would be helpful to classify exploits according to this second difficulty level. When an exploit exists, no matter how difficult it was to develop (first

level of difficulty) then it can be bought and sold, or given away. That is, it can be made available for anyone to use; and this is a social condition, not an engineering condition. However, the second level of difficulty may well hinder some intruders who wish to distort the normal operation of S. When an exploit exists, if the malaction is external then it may be directly activated. For external malactions, srv-level 3 is maximal. If the associated malaction is

internal, then its associated sr-vulnerability cannot be activated in S until the exploit is installed in S. If there is no means of introducing the exploit into S (say S is rigorously "air-gapped", with rigorously-enforced privileged access and no means of allowing external input) then the sr-vulnerability cannot be activated. So srv-level 3 is not maximal for internal malaction.

**SRV-Level 4** An internal exploit exists for the sr-vulnerability, and a vector exists for S; that is, a means exists to introduce the exploit into system S. Then system S can be attacked through the vulnerability by using the vector to introduce the exploit into S. Whether this actually occurs is dependent largely upon social parameters rather than engineering parameters. I say the vulnerability is srv-level 4. The actual threat associated with a srv-level 4 sr-vulnerability depends on social parameters.

**SRV-Level 5** An internal exploit is installed in S. The vector has been used. The exploit is in position to activate the vulnerability. All that is necessary for this activation is for the exploit to execute. Srv-level 5 is the maximal automatic malaction level.

The srv-levels can be thought of as barriers to activation of a sr-vulnerability in $S$, to cause $S$ to relinquish some of its required safety properties. At srv-level 1, a vulnerability must be discovered, an exploit found, and then a vector. At srv-level 2, a known sr-vulnerability lacks an exploit and, if it is to be internally exploited, a vector for it to be installed and activated. At srv-level 3, an exploit exists but, if it is internal, not yet a vector for installation. At srv-level 4, the internal exploit is installed and ready to activate the sr-vulnerability.

It may be asked what the point is of defining srv-level 1: surely, what you don't know can't hurt you. However, it represents a concrete possibility of future harm; not immediately, but when its level is eventually raised through discovery and exploitation. Just as there are helpful ways of estimating how many undiscovered bugs there are in deployed systems, there may be ways, maybe similar or the same ways, which can be used to estimate how many undiscovered sr-vulnerabilities there are likely to be in a deployed system; that is, the number of srv-level 1 sr-vulnerabilities it contains. It is useful to have a name for those, namely srv-level 1.

I define the severity of a sr-vulnerability to be the severity of the hazard with which the sr-vulnerability is associated. Values of a severity are given by how the values

are defined in the risk analysis of the system S required by a standard conforming to Guide 51. In case the sr-vulnerability is associated with more than one hazard, the severity is the maximal severity of any of the hazards with which the sr-vulnerability is associated.

I define the safety-related security risk (sr-security-risk or srs-risk) of a sr-vulnerability to be the combination of sr-level with severity of the sr-vulnerability, along with the classifications of malactions as induced or automatic and external or internal. The risk is intuitively the possibility of activation. In the terms of Clarke, it is a possibilistic or possibility-based concept, in contrast to that of safety-related risk which is probability-based [Cla06]. It depends on what is available. There might be more than one sr-vulnerability, correspondingly more than one piece of malware, associated with a hazard; such malware might induce external or internal malaction; and if it is external then srv-level 3 is already the lowest barrier level/highest technical-threat level, whereas for internal malactions two more levels represent further-reduced barriers/higher technical threat. So possibilistic risk may best be expressed as divided into external and internal threats, with the correspondingly highest srv-level. Further, an induced internal malaction is associated with a further barrier, namely a person must have appropriate access to S in order to induce the malaction. For this reason, I also consider automatic malaction to represent a higher threat than induced malaction.

I propose that srs-risk be represented as an ordered triple containing two triples:

$$<< \text{external}, [\text{induced/automatic}], \text{srv}-\text{level} >,$$

$$< \text{internal}, [\text{induced/automatic}], \text{srv}-\text{level} >, \text{severity} >$$

where

- [induced/automatic] = automatic if there is an automatic associated malaction,
- [induced/automatic] = induced if there is an induced associated malaction but no automatic associated malaction,
- [induced/automatic] is omitted if there is no associated malaction,

The srs-risk of an sr-vulnerability indicates the ease, respectively difficulty, of activating the vulnerability.

This conception of sr-security-risk is somewhat different from the conception of risk used in safety analysis in that it is possibilistic. The safety conception of risk is probabilistic; some estimate is required of the likelihood that a hazard will occur, and that occurrence of the hazard will lead to actual harm. This estimate is then combined with the severity of the hazard to constitute safety risk. However, as argued in [Lad16.1,Lad16.2], the likelihood of activation of a vulnerability varies largely with social factors, not engineering factors, and those social factors can vary considerably with time without the engineering situation changing at all. I consider it inappropriate for an engineering conception to include variable social factors with little connection to engineering concepts. The possibilistic approach is more appropriate for sr-security-risk.

A guide for assessment of sr-security risk must be adapted to the possibilistic approach outlined. I suggest it proceed as follows.

**Safety-Related Vulnerability Identification** Hazards can be thought of as situations which are precursors to harm. Malicious intrusion can induce the occurrence of a hazard. A safety-related vulnerability is a means by which a hazard can be induced through malicious intrusion.

**SR-Security-Risk Analysis** The srv-level of a sr-vulnerability is assessed. If the srv-level is sufficiently high, then classification as external/internal shall occur, as well as classification as induced/automatic.

**SR-Security-Risk Assessment** The srs-risk is calculated as above.

**Risk Reduction** Measures must be introduced into the system to reduce the srs-risk to an acceptable level. Risk reduction measures include:

**Eliminating sr-vulnerabilities ("patching")**

- Introducing additional safety functions (reducing the severity of an sr-vulnerability);
- Adding intrusion-detection systems internal to S (prophylaxis against internal induced malaction);
- Adding malware-detection systems internal to S (prophylaxis against internal automatic malaction);
- Introducing system-external srs-risk control measures (e.g., human monitors of activity in S and around S)

## 14.6 Definitions and Terms

**activate (a sr-vulnerability):** execution of an exploit for a sr-vulnerability

**anomalous execution:** a system behaviour which does not conform to its designed and specified functionality

**associated:** an sr-vulnerability V which induces a hazard H is said to be associated with H

**automatic malaction:** malaction caused through autonomous anomalous execution when a certain system state is attained

**exploit:** software code, firmware, system-modification behaviour (internal exploit) or anomalous environmental behaviour (external exploit) which induces the hazard associated with an sr-vulnerability

**external malaction:** exploitation of service mechanisms which the system S offers to its environment, without modification of system S. In response to external malaction, system S responds with its designed and specified functionality

**induced malaction:** malaction conducted by direct action of a human being

**install:** use of a vector to introduce an internal exploit into a system

**internal malaction:** malaction achieved through modification of the internal state space of system S itself, through, say, insertion of new software code, or modification of hardware.

**malaction:** anomalous execution or anomalous environmental behaviour induced through malicious action

**safety-related vulnerability (sr-vulnerability):** a means by which a hazard H can be causally induced through malicious action (malaction)

**safety-related security risk (sr-security risk, srs-risk):** The triple < <external, [ind/aut], srv-level>, <internal, [ind/aut], srv-level>, severity> where

[ind/aut] = automatic if there is an automatic associated malaction, [ind/aut] = induced if there is an induced associated malaction but no automatic associated malaction, and [ind/aut] is omitted if there is no associated malaction

**srv-level:** one of five defined levels which indicate the barriers, or lack of barriers, to activation of a sr-vulnerability through an exploit

**vector:** a means through which an internal exploit is introduced into a system

**vulnerability:** a mechanism $V$ allowing $S$ to enter a state *St-bad* from which $S$ causes the system to behave in a way which results in harm

# Bibliography


[1] Lee Clarke, *Worst Cases*, University of Chicago Press, 2006.

[2] International Electrotechnical Commission, *IEC 60050 International Electrotechnical Vocabulary*, no date. Available at http://www.electropedia.org, accessed 2017-11-16.

[3] International Standards Organisation/International Electrotechnical Commission, *ISO/IEC Guide 51 Edition 3, Safety aspects – Guidelines for their inclusion in standards*, 2014.

[4] International Electrotechnical Commission, Project AC/18/2017 of 2017-06-02, *Draft Guide 120 Edition 1, Security aspects – Guidelines for their inclusion in standards*. Distributed as DKE Rundschreiben INFO_2017-0026 on 2017-06-07.

[5] Peter Bernard Ladkin, *Risks People Take and Games People Play*, in Parsons, M. and Anderson, T., *Engineering Systems for Safety*, Proceedings of the Twenty-Third Safety Critical Systems Symposium, SSS 2015, Bristol, UK, 3-5 February 2015, ISBN 978-1505689082, SCSC on Amazon. Available from https://rvs-bi.de/publications/Papers/SSS2015LadkinPublFinal.pdf , accessed 2017-05-24.

[6] Peter Bernard Ladkin, *Risks People Take and Games People Play:Talk*, slideset to accompany the invited talk on [5] at the 23rd Safety-Critical Systems Symposium, SSS 2015, Bristol, 3-5 February 2015. Available from https://rvs-bi.de/publications/Talks/LadkinSSS2015.pdf , accessed 2017-05-24.

[7] Peter Bernard Ladkin, *References to Cybersecurity in the Functional Safety Standards IEC 61508:2010 and 61511:2016*, 2017. Preprint, reprinted in this volume.

[8] Ingo Rolle, *personal communication*, 2017-05-23.