International Journal of Comparative Studies in International Relations and Development
p-ISSN: 2354-4198 | e-ISSN: 2354-4201
Volume 10, Number 1 May, 2024

IJCSIRD

Yet Another Cybersecurity Risk Assessment Framework

[1]Mathias Ekstedt, [2]Zeeshan Afzal, [3]Preetam Mukherjee,
[4]Simon Hacks & [5]Robert Lagerström
[1,2&5]KTH Royal Institute of Technology, Stockholm, Sweden
[3]Digital University Kerala, Thiruvananthapuram, India
[4]Stockholm University, Stockholm, Sweden

Article DOI: 10.48028/iiprds/ijcsird.v10.i1.07

Abstract

IT systems pervade our society more and more, and we become heavily dependent on them. At the same time, these systems are increasingly targeted in cyberattacks, making us vulnerable. Enterprise and cybersecurity responsible face the problem of defining techniques that raise the level of security. They need to decide which mechanism provides the most efficient defense with limited resources. Basically, the risks need to be assessed to determine the best cost-to-benefit ratio. One way to achieve this is through threat modeling; however, threat modeling is not commonly used in the enterprise IT risk domain. Furthermore, the existing threat modeling methods have shortcomings. This paper introduces a metamodel-based approach named Yet Another Cybersecurity Risk Assessment Framework (Yacraf). Yacraf aims to enable comprehensive risk assessment for organizations with more decision support. The paper includes a risk calculation formalization and also an example showing how an organization can use and benefit from Yacraf.

Keywords: *Threat modeling, Enterprise IT risk, Risk assessment, Attack tree*

Corresponding Author: Mathias Ekstedt

First Published: https://doi.org/10.1007/s10207-023-00713-y

Background to the Study

With the increased pervasiveness and complexity of the IT infrastructure induced by the digitization, the importance of cybersecurity management also increases. From an enterprise management point of view, the cybersecurity responsibility has landed on its own role, the chief information security officer (CISO), and largely falls under the challenge of risk management. Bottom line, the CISO must determine what security controls should be applied in an IT infrastructure so that business risks and costs are minimized at the same time. In parallel to the developments in the enterprise cybersecurity management domain, the software engineering community has witnessed a corresponding increase in attention on the topic of security, resulting in the emergence of a community around the concept of threat modeling. This movement is perhaps most clearly represented by Microsoft's work on developing the secure development life cycle (SDL) in its organization during the early 2000s in combination with Shostack's book, which describes the Stride method.

In particular, our goal with this work is to merge two strongholds from the two communities: the model-based security analysis from the threat modeling community and the quantitative risk assessment calculations from the risk management community. We set the following concrete objectives for this work.

1. To propose a metamodel for risk-based threat modeling.
2. To provide a risk calculation framework.

The most prominent work with a somewhat similar agenda is found in the method Process for Attack Simulation & Threat Analysis (PASTA). The presented approach is named Yet Another Cybersecurity Risk Assessment Frame-work (Yacraf) and the novelty in this approach is that we combine a metamodel with tailored logic for risk assessment calculations into a unified framework. This enables us to take the structure and architecture of IT systems and their context into account in the risk assessment.

In general, our approach adheres to the common view that *risk* is a function of *threat*, *vulnerability*, and *impact*, found for instance in FAIR and PASTA. However, in other details, our Yacraf metamodel differs. The presented metamodel provides transparency in how to argue around the value of different parameters in the risk assessment equation.

Related Work

This section presents the related work. As the goal of this work is to integrate model-based security analysis from the threat modeling community with quantitative risk assessment calculations from risk management, the section begins with an overview of various threat modeling and risk assessment methods currently available.

Threat Modeling and Risk Assessment Methods

A multitude of methods exist for conducting threat modeling and risk assessment and management. STRIDE considers possible threats while a product or system is under development. The method involves creating a model of the system using data flow diagrams

(DFDs) and then considers different threats that can impact each part of the model. The threats are generally known and relate to the method name, STRIDE, which stands for Spoofing identity, Tampering with data, Repudiation, Information disclosure, Denial of service, and Elevation of privilege. DREAD is a modified STRIDE approach developed by Microsoft to evaluate threats. It refers to five categories; Damage potential, Reproducibility, Exploitability, Affected users, and Discoverability. It proposes a different method for threat assessment where values are assigned to different categories, allowing for an average value to be calculated to represent the overall risk.

Comparison of Methods

The decision of which method to be use for a particular organization or system is a complex task that is further complicated by the fact that there is no perfect method. Different methods are designed with different points of view and often address different goals. To motivate the need for another model-based risk assessment framework, as proposed in this paper, we compare a number of existing methods. It is not possible to cover all possible methods; hence, the scope is set to the most well-known methods and the list of such methods is inspired by previous works. In total, seven approaches are considered. These are compared with each other using three different indicators, as discussed below. It should be noted that the presented comparison is merely a collation of related work and not a systematic literature review.

The first indicator is *scope* and is inspired by previous work in. By *scope*, we mean the perspective or point of view for which the method was developed. *Scope* is further defined by two sub indicators, namely; *approach* and *goal*. *Approach* refers to the focus of a method, as different methods might have unique strategies. Some are focused on the design stage or the architecture of a system, while others on modeling IT assets or business objects, attackers, and the consequences. The method's *goal* could also differ and be of importance. Table compares the different methods based on this indicator.

The second indicator is the *level of detail* (complete-ness), or the depth covered by a framework. As mentioned earlier, *risk* is commonly considered to be a function of *threat*, *vulnerability*, and *impact*. Therefore, a comprehensive risk analysis framework should include all three conceptual domains. However, different methods apply modeling at different granularity and stages. Some methods only focus on vulnerabilities and perform modeling of either the business objects and/or the IT assets, while others consider the threat actor and the impact too. By asset modeling, in this context and the rest of the paper, we mean identifying and modeling (all) the IT assets of the organization's infrastructure and their internal communications to identify vulnerabilities and attack surfaces. Another group of methods model threat actor by representing potential attackers to the organization and modeling their abilities and possible attack vectors. Finally, potential loss from possible attacks in a specific organizational scenario is estimated by some methods as part of impact or consequence modeling.

The third and final indicator is the *type of assessment* employed by a method. Some methods allow numeric and quantitative risk assessment, while others only enable a qualitative assessment. It should be noted that a quantitative risk assessment can still include some underlying qualitative input parameters. Another important differentiation between different models is whether they provide an explicit and consistent metamodel for cyber-risk assessment. Table shows the comparison of different methods based on indicator two and three and their relevant sub indicators.

This motivates the need for another approach, as described in this paper. This approach named Yet Another Cyber Security Risk Assessment Framework (Yacraf) will allow to enrich the enterprise IT risk domain by the means of threat modeling. Yacraf leverages the benefits and combines the two domains of model-based security analysis and quantitative risk assessment. Furthermore, it goes one step further and integrates a model-based quantitative risk assessment in an explicit metamodel that provides more decision support than any other existing method. The result is also expected to be more realistic as the risk assessment provides additional resolution by considering the structure and architecture of IT systems and their contexts. Yacraf is comprehensive and is intended to be used by end-user IT organizations as a tool in their cyber risk analysis. To put Yacraf into context, we classify it according to the three presented indicators (*Scope*, *Level of detail*, and *Type of assessment*) above. Yacraf's goal is to provide a quantitative risk assessment with an approach that is focused on different IT components (assets). More-over, it covers all three conceptual domains and enables modeling for assets, threats, and the impact. Finally, Yacraf provides a metamodel and a quantitative assessment.

Table 1: Comparison according to indicator 1 - Scope

Framework	Reference	Approach	Goal
STRIDE	[38]	Software-centric	Identify threats
DREAD	[37]	Software-centric	Evaluate threats
OCTAVE	[6]	Asset-centric	Organizational risk
ISMS-/CORAS	[5, 21]	Asset-centric	Risk assessment
FAIR	[10]	Asset-centric	Risk assessment
PASTA	[25]	Risk-centric	Risk assessment
TRIKE	[32]	Asset-centric	Risk assessment

Table 2: Comparison According to Indicator 2 and 3—*Level of detail* and *type of assessment*

Method	Business object modeling	Asset modeling	Threat actor modeling	Impact modeling	Metamodel	Quantitative	Qualitative
STRIDE		✗					
DREAD		✗					✗
OCTAVE	✗		✗	✗			✗
ISMS-/CORAS	✗	✗	✗	✗	✗	✗	✗
FAIR		✗	✗	✗		✗	
PASTA		✗	✗	✗		✗	✗
TRIKE		✗	✗	✗			

The Framework

In this section, we present our proposed framework. It con-sists of a metamodel that defines what needs to be modeled in order to perform cybersecurity risk assessment, as well as a formalism for deriving the risk assessment from a model instance.

Metamodel

The metamodel describes classes, their associations, and class attributes. It is conceptually divided into the three risk assessment domains: *vulnerability*, *threat*, and *impact*.

Vulnerability

The Asset is the central element of our metamodel. The Asset is used as an abstract class representing any kind of IT related component. The idea is then to specialize Assets into any class that make sense for the system domain at hand. Since this will vary, we here introduce exemplary classes that have been inspired by (but not identical to) DFDs often found in the threat modeling community. Moreover, when Assets are refined, additional class associations are (normally) added. In our metamodel, we do not elaborate on all possible associations we could or would like to include for different sub assets. Instead, we introduce an Asset self-association as a place holder for all of these. Our ambition is thus to be flexible with the exact design of the metamodel for system modeling, because this will vary in practice.

In order to be able to assess the vulnerability dimension of the risk assessment, we provide the class Vulnerability, which is related to Asset. A Vulnerability can be exploited, which we describe by an Attack event, but also protected with Defense mechanisms. Generally, a Vulnerability can be under-stood as a composition of Attack events and Defense mechanisms. In turn, this also means that the relation between Vulnerability and Asset is a derived relationship depending on the Attack event and Defense mechanism relationships to the Asset.

Threat

Attack events are executed by an Attacker. To express the planning behind attacks, we facilitate Abuse cases. Analogous to use cases, an abuse case is a set of actions representing some complete system interaction, but as opposed to use cases with a malicious intent. In our metamodel, these actions are the attack events; thus, an Abuse case is a composition of a number of Attack events. As the Attack events are ordered in graphs, some of them would constitute the attack surface (the Attack events without parents) and there would be at least one end goal (the Attack events without children).

Impact

To understand the impact of a risk, the concept of Loss events is crucial. In general, the discussion of causes and consequences within the risk analysis field is diverse. Here, to approach this distinction with the assumption that Attack events are happening to the IT or cyber domain and Loss events relate to the business or physical context that the cyber-Asset is connected to. An Attack event is thus causing a Loss event. For instance, the confidentiality breach of some customer records is an attack that lacks any inherent consequence; this is instead captured as a Loss event such as regulatory fines, lost reputation, or customer privacy exposure.

Identification of vulnerabilities on assets enables us to identify possible attack events. Various attack events can be causally related and can form attack graphs. The attack events are marked in colors to show their correspondence with the assets in Defenses on the organizational assets can be used to protect vulnerabilities from exploitation.

Table 3: List of vulnerabilities

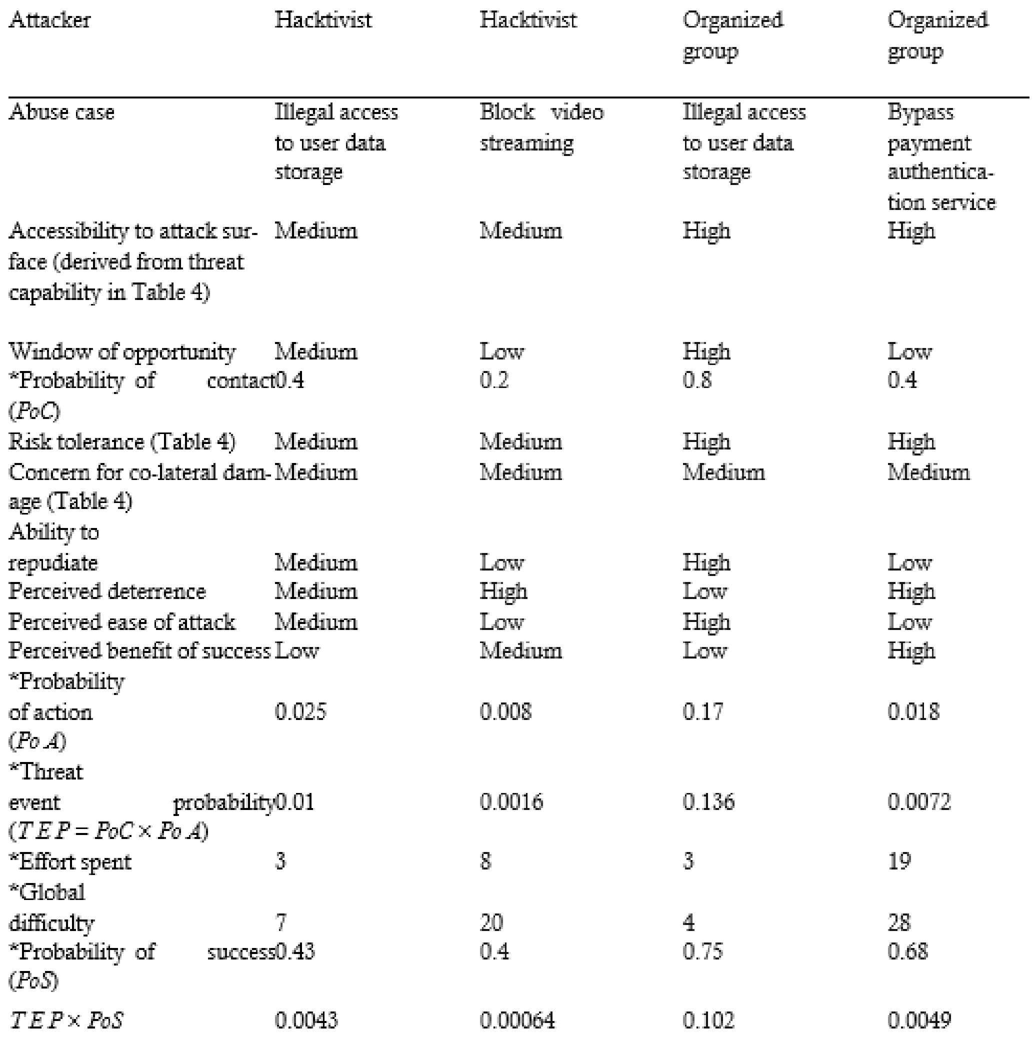
Asset	Vulnerability	Severity Defense
Authentication service	Authentication bypass by spoofing (CWE-290)	Medium Multi-factor authentication (M1032)
User data storage (MySQL 5.5.0)	Buffer overflow (CVE-2013-1492, CWE 119)	High Update software (M1051)
Video storage (MongoDB Server v4.4)	Read Overrun (CVE-2020-7928)	Medium Update software (M1051)
Video streaming service (nginx v1.15.5)	Memory disclosure (CVE-2018-16845)	Medium Restrict file and directory permissions (M1022)
Payment authentication service	Origin validation error (CWE-346)	High Change software configuration (M1054)
Payment data storage (postgreSQL 9.5.2)	NULL pointer dereference (CVE-2016-5423, CWE-476)	Medium Update software (M1051)

We have identified some use cases related to the business goals. These use cases are implemented by company assets and actors. Potential loss events caused by attack events are presented in Table . Impacted use cases and suffered actors due to the loss events are also listed in the table. Company is treated as an actor suffering from all the loss events.

Table 4: Attacker characteristics

Attacker	Risk tolerance	Concern for collateral damage	Skill	Resources	Sponsorship	Derived threat capability
Hacktivist	Medium	Medium	Medium	Medium	Low	Medium
Organized crime group	High	Medium	High	High	Medium	High

Table 5: Abuse Cases



Attacker	Hacktivist	Hacktivist	Organized group	Organized group
Abuse case	Illegal access to user data storage	Block video streaming	Illegal access to user data storage	Bypass payment authentication service
Accessibility to attack surface (derived from threat capability in Table 4)	Medium	Medium	High	High
Window of opportunity	Medium	Low	High	Low
*Probability of contact (PoC)	0.4	0.2	0.8	0.4
Risk tolerance (Table 4)	Medium	Medium	High	High
Concern for co-lateral damage (Table 4)	Medium	Medium	Medium	Medium
Ability to repudiate	Medium	Low	High	Low
Perceived deterrence	Medium	High	Low	High
Perceived ease of attack	Medium	Low	High	Low
Perceived benefit of success	Low	Medium	Low	High
*Probability of action (PoA)	0.025	0.008	0.17	0.018
*Threat event probability ($TEP = PoC \times PoA$)	0.01	0.0016	0.136	0.0072
*Effort spent	3	8	3	19
*Global difficulty	7	20	4	28
*Probability of success (PoS)	0.43	0.4	0.75	0.68
$TEP \times PoS$	0.0043	0.00064	0.102	0.0049

$*$notation indicates the computed parameters

Risk Assessment

Risk values computed by the Yacraf risk assessment method proposed in this paper are shown in Table . Estimated loss event magnitude values are multiplied with the loss event probability values (derived from Table) to compute the actor and loss event specific risk. Total risk or actor specific risk can be computed by summing up the individual risk values.

While computing the *Local difficulty* values, it is assumed that all the defense mechanisms are active. With all activated defenses, the Calculated Risk on the actor *Company* for the loss event *Videos unavailable* will be e3200. If there are no active defenses in the attack graph, the local difficulty values for the attack events

a) Bypass authentication_service_by_spoofing,

b) Buffer overflow_on_user_data_storage
c) Memory_disclosure_in_video_streaming_service will become 5, 2, and 5, respectively.

The *Global difficulty* value for the tar-get attack event, i.e., *Stop_streaming_service*, will become 12. In this scenario, the Calculated Risk on the actor *Company* for the loss event *Videos_unavailable* will be e5500. At this point, if *Restrict_file_and_directory_permissions* defense mechanism is activated, the global difficulty for the *Stop_streaming_service* will become 17. Calculated Risk will become e3750. Any risk assessment method should satisfy the general intuitions about risk.

Table 6: Loss events

Target attack event	Access user data	Stop streaming service	Stop streaming service	Change ment data
Loss event	User data leaked	Videos unavailable	Videos unavailable	Payment data changed
Impacted use cases	Accessing videos	Streaming videos	Streaming videos	Making payments
Loss event type	Financial loss, legal issues	Productivity loss, loss of reputation, loss of competitive advantage	Productivity loss, financial loss	Financial loss, legal issues, loss of reputation
Suffered actor Magnitude Abuse case	Company e100,000 Illegal access to user data storage (by Hacktivist and Organized group)	Company e5,000,000 Block video streaming (by hacktivist)	User e1,000 Block Video streaming (by hacktivist)	Company e10,000,000 Bypass Payment authentication service Organized group)
*Loss event probability (*L E P*) (from Table 5)	0.107	0.00064	0.00064	0.0049
*Risk (*L E P* × *magnitude*)	e10,700	e3200	e0.64	e49,000

*Notation indicates the computed parameters

Conclusion

This paper presents a model-based risk assessment approach named Yet another cybersecurity risk assessment frame-work (Yacraf). Yacraf allows a holistic risk assessment for organizations by combining the two domains of model-based security analysis and quantitative risk assessment. The core novelty of this approach, however, revolves around the introduction of an explicit metamodel for model-based cyber-security risk assessment. This enables more transparent and structured decision support than other approaches. The paper includes a formalization of risk calculations and also an example instant of how an organization can make use of Yacraf. The paper also provides a short summary of practical experiences of using Yacraf in real-world organizations in case studies. These studies demonstrate the positive potential of using Yacraf.

Recommendation

The Study recommended that *Global difficulty* value for the target attack event *multi-factor_authentication* is the only activated defense mechanism, If global difficulty for the *Stop_streaming_service,* Let us assume no defense mechanisms are enabled. As given above, the *Global difficulty* value for the target attack event will remain unaltered,

References

Alam, M., Breu, R., Hafner, M. (2007). Model-driven security engineering for trust management in secret*, J. Softw. 2*(1), 47–59

Basin, D., Doser, J. & Lodderstedt, T. (2006). Model driven security: From uml models to access control infrastructures, *ACM Trans. Softw. Eng. Methodol. (TOSEM) 15*(1), 39–91

Caralli, R. A., Stevens, J. F., Young, L. R., & Wilson, W. R. (2007). Introducing octave allegro: improving the information security risk assessment process. Carnegie-Mellon Univ Pittsburgh PA Software Engineering Inst, Technical report

Deng, M., Wuyts, K., Scandariato, R., Preneel, B., & Joosen, W. (2011). A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements, *Requir. Eng. 16*(1), 3–32

Freund, J., & Jones, J. (2015). Measuring and managing information risk. Butterworth-Heinemann, Waltham https://doi.org/ 10.1016/C2013-0-09966-5

Friman, N. (2020). Security analysis of smart buildings. Bachelor thesis, School of Electrical Engineering and Computer Science (EECS), KTH Royal Institute of Technology

Hafner, M., Breu, R., Agreiter, B., & Nowak, A. (2006). Sectet: An extensible framework for the realization of secure inter-organizational workflows, *Nternet Res. 16*(5), 491–506

Johnson, P., Lagerström, R., & Ekstedt, M. A. (2018). *Meta language for threat modeling and attack simulations*. In: Proceedings of the 13th International Conference on Availability, Reliability and Security, 1–8

Jürjens, J. (2005). *Secure systems development with UML*. Springer, Berlin, Heidelberg

Kordy, B., Mauw, S., Radomirovi, C, S., & Schweitzer, P. Foundations of attack–defense trees. In: International Workshop on Formal Aspects in Security and Trust, 80–95

Lund, M. S., Solhaug, B., & Stølen, K. (2010). *Model-driven risk analysis: The CORAS approach*, Springer, Berlin, Heidelberg

Mathey, F., Bonhomme, C., Rocha, J., Lombardi, J., & Joly, B. (2018*). Risk assessment optimization with MONARC*, https://www.monarc.lu/assets/files/publications/2018-HACK.LU-CASES.pdf

Morana, M. M., & Uceda, V. T., (2015*). Risk centric threat modeling: Process for attack simulation and threat analysis,* Wiley, Hobo-ken, New Jersey

Paja, E., Dalpiaz, F. & Giorgini, P. (2015). Modelling and reasoning about security requirements in socio-technical systems, *Data Knowl. Eng. 98*, 123–143

Potteiger, B., Martins, G., & Koutsoukos, X. D. (2016). *Software and attack centric integrated threat modeling for quantitative risk assessment. In: Scherlis WL, Brumley D (eds*) Proceedings of the Symposium and Bootcamp on the Science of Security, Pittsburgh, April 19–21, 2016, 99–108. ACM

Shevchenko, N., Chick, T. A., O'Riordan, P., Scanlon, T. P., & Woody, C. (2018*). Threat modeling: A summary of available methods*, Carnegie Mellon University Software Engineering Institute, Technical report

Shostack, A. (2008). *Experiences threat modeling at Microsoft*, Technical report, Microsoft

Tuma, K., Calikli, G., & Scandariato, R. (2018). Threat analysis of software systems: A systematic literature review, *J. Syst. Softw. 144,* 275– 294

Wang, J., Neil, M., & Fenton, N. E. (2020). A Bayesian network approach for cybersecurity risk assessment implementing and extending the FAIR model, *Comput Secur 89*

Wessman, L., & Wessman, N. 2020). Threat modeling of large-scale computer systems: implementing and evaluating threat modeling at company x. Bachelor thesis, School of Electrical Engineering and Computer Science (EECS), KTH Royal Institute of Technology

Xiong, W. & Lagerström, R. (2019). Threat modeling—a systematic literature review, *Comput. Secur. 84*, 53–69

Xiong, W., Hacks, S., & Lagerström, R. A. (2021). Method for assigning probability distributions in attack simulation languages, *Complex Syst. Inf. Model Q 26,* 55–77