Cyber Threat Modeling: An Evaluation of Three Methods

FORREST SHULL AND **NANCY R. MEAD**

NOVEMBER 11, 2016

CITE

Get Citation❞

TAGS

Cyber Missions | Threat Modeling

Cyber threat modeling, the creation of an abstraction of a system to identify possible threats, is a required activity for DoD acquisition. Identifying potential threats to a system, cyber or otherwise, is increasingly important in today's environment. The number of information security incidents reported by federal agencies to the U.S. Computer Emergency Readiness Team (US-CERT) has increased by 1,121 percent from 5,503 in fiscal year 2006 to 67,168 in fiscal year 2014, according to a 2015 Government Accountability Office report. Yet, our experience has been that it is often conducted informally with few standards. Consequently, important threat scenarios are often overlooked.

Given the dynamic cyber threat environment in which DoD systems operate, we have embarked on research work aimed at making cyber threat modeling more rigorous, routine, and automated. This blog post evaluates three popular methods of cyber threat modeling and discusses how this evaluation will help develop a model that fuses the best qualities of each.

The State of Cyber Threat Modeling

In addition to being a requirement for DoD acquisition, cyber threat modeling is of great interest to other federal programs, including the Department of Homeland Security and NASA. When cyber threat modeling is applied to systems being developed it can reduce fielded vulnerabilities and costly late rework. However, there are challenges in the existing approaches.

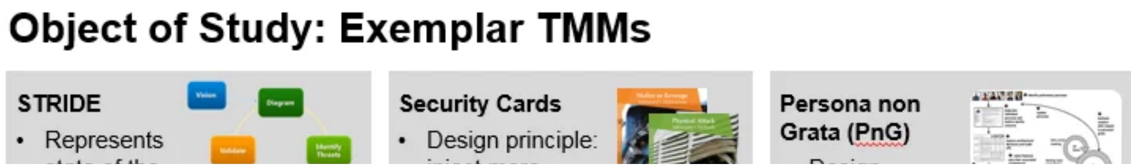
One challenge that we have seen with threat modeling is that it asks engineers to put themselves in a mindset that they aren't often asked to take. In particular, engineers focus largely on building a system and meeting functionality requirements. It is hard, therefore, for them to change hats and envision potential threats.

We coordinate a working group on cyber threat modeling for the DoD. This working group is a forum for organizations doing threat modeling to share their experiences and challenges. Looking across organizations, it is clear there are different types of modeling being used, not always with much in common. For example, it is an open question--and a point of disagreement among different organizations--whether modeling types of attackers and their capabilities can be helpful for identifying what types of threats will be important for a given system. One question, expressed in different ways, frequently debated is:

What abstraction of the system will give the greatest insight into what the cyber threats are?

An Evaluation of Three Methods

In the first phase of our research, we examined three "exemplar" approaches--STRIDE, Security Cards, and Persona non Grata--to measure how they worked in practice. These exemplar approaches were selected because they incorporate different modeling strategies that are often discussed. We created an operational concept description for two small systems common in the DoD: a drone system and an IT system for aircraft maintenance data.



Object of Study: Exemplar TMMs

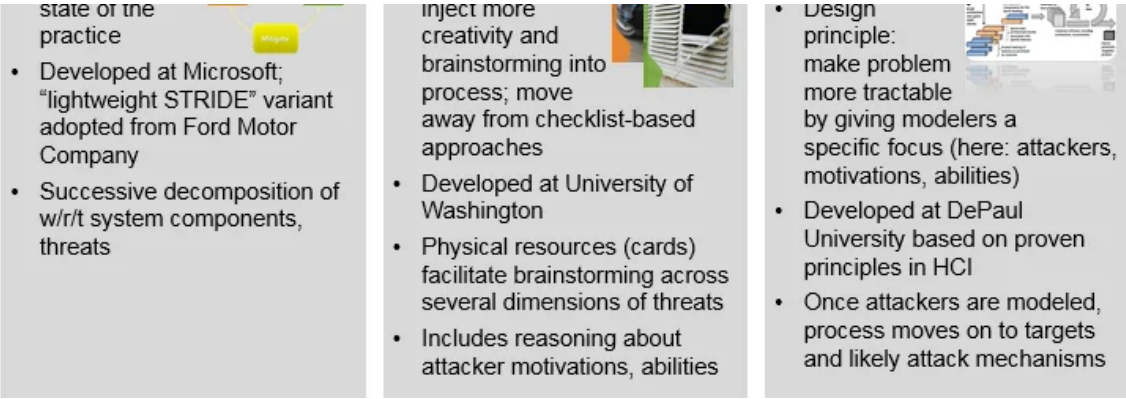
STRIDE

- Represents

Security Cards

- Design principle:

Persona non Grata (PnG)



state of the practice
- Developed at Microsoft; "lightweight STRIDE" variant adopted from Ford Motor Company
- Successive decomposition of w/r/t system components, threats

inject more creativity and brainstorming into process; move away from checklist-based approaches
- Developed at University of Washington
- Physical resources (cards) facilitate brainstorming across several dimensions of threats
- Includes reasoning about attacker motivations, abilities

- Design principle: make problem more tractable by giving modelers a specific focus (here: attackers, motivations, abilities)
- Developed at DePaul University based on proven principles in HCI
- Once attackers are modeled, process moves on to targets and likely attack mechanisms

We then worked with our university partners--DePaul University, University of Washington, and The University of Utah--who agreed to teach the three threat modeling approaches to over 250 subjects of varying experience and education levels, from the fields of cybersecurity and software engineering. Subjects worked in teams to apply different threat modeling approaches on each of the two systems.

We tracked data to compare the performance of the approaches on the following factors, comparing the results of the 3 methods to one another *and* to analysis results by experts:

- number of threat types detected
- number of threats missed
- number of false positives reported

Specifically, we tried to identify tradeoffs among the three methods, as well as a degree of confidence that users can expect from each of the three methods. As described in the remainder of this post, no single approach outperformed the others across the board. Rather, we found that the best approach depends on the system and environmental context in which it is used.

STRIDE. STRIDE is an acronym consisting of the following six categories:
-spoofing identity
-tampering with data
-repudiation
-information disclosure
-denial of service
-elevation of privilege

STRIDE was developed at Microsoft and represents the state of the practice (a lightweight variant of STRIDE, for instance, was adopted by the Ford Motor Company). STRIDE involves modeling a system and subsystem and how data flows through the system and subsystem. After that, the methodology relies on a checklist evaluation approach based on the six categories listed above.

Subjects who used the STRIDE method did not report a lot of false positives, but the teams generally obtained inconsistent results. The threats reported seemed to have more to do with the makeup of specific teams and their background or experience.

Based on our initial analysis, STRIDE seems an ideal approach for teams that don't have a lot of security expertise because the checklist-based approach constrains users and limits the potential for false positives. One weakness of STRIDE, however, is that it is an onerous task to apply checklists of potential threats to the components of the various systems and subsystems.

Security Cards. The Security Cards approach moves away from checklist-based approaches like STRIDE and injects more creativity and brainstorming into cyber threat modeling. The motivation behind this approach is that it can help users identify unusual or more sophisticated attacks. Developed at the University of Washington, the Security Cards method relies on physical resources (i.e., cards) to facilitate brainstorming about potential cyber threats. Subjects were also asked to include reasoning about attacker motivations and abilities.

With Security Cards we found that, overall, the teams of participants exhibited higher effectiveness. Almost all types of threats were found by teams using Security Cards, but the Security Cards approach also exhibited greater variability across teams. This approach, however, produced many false positives. The high number of false positives makes sense because users are encouraged to brainstorm and come up with unusual or atypical scenarios. Similarly, the performance across teams was dissimilar. By applying Security Cards, there weren't many threats that the teams couldn't eventually identify, but each team only found a subset of threats, and that subset varied substantially from team to team.

Given our initial results, Security Cards would seem to be an ideal approach in scenarios where a user values a wider spectrum of results over consistent results.

Persona Non Grata. Developed at DePaul University, the Persona non Grata approach makes threat modeling more tractable by asking users to focus on attackers, their motivations, and abilities. Once this step is completed, users are asked to brainstorm about targets and likely attack mechanisms that the attackers would deploy.

The theory behind this approach is that if engineers can understand what capabilities an attacker may have, and what types of mechanisms they may use to compromise a system, the engineers will gain a better understanding of targets or weaknesses within their own systems and the degree to which they can be compromised.

Some critics of this approach argue that Persona non Grata can often take users down the wrong path. For example, for a system related to national security, users might reason that the system may be the target of a sophisticated attack from another nation state. This conclusion, however, overlooks the fact that a nation state might compromise a system first through a much simpler entry point and then ratchet up operations from there.

With Persona non Grata, our research participants reported fewer false positives, but they also were unable to gain a comprehensive view of potential threats. Their threat modeling tended to consistently produce only a subset of threat types, which we identified as a drawback to this approach.

While the teams using Persona non Grata did not identify all the threats, the threats they did identify were reproduced consistently across teams. This is important if the aim of threat analysis is to identify a potential threat (within that subset) with a [high?] degree of confidence. Moreover, if a threat modeler has more awareness of the types of vulnerabilities that are important in a system, Persona non Grata is ideal because it gives the user a higher degree of confidence in his or her ability to identify priority threats.

Seeking Future Collaborators