Which Threat Modeling Tool is Right for You?

Microsoft TMT vs. ThreatModeler™

by Reef Dsouza, Security Consultant at Amazon Web Services

Ubiquitous cyber attackers pose constant challenges to even the most robust security fortifications. They add a plethora of new threats daily to the cyber-ecosystem. Cybersecurity can no longer be just another cost of doing business. Senior executives are increasingly considering InfoSec and OpSec as strategic business components. This is giving rise to significant increases in security budgets. Market analysts expect the cyber security market value to top $201.36 billion by 2021.[i] To date, though, no matter how much organizations beef up their security defenses and big-data analytics capacity, it does not seem to make a difference. Malicious actors find a way through the defenses and go undetected by the analytics. Furthermore, attacks which at one time were considered complex, requiring the resources and commitment of large-scale organized crime or nation-states, are now possible with freely available, automated exploit tools. As long as organizations take a defensive posture with their IT security, they relinquish the initiative to attackers.

The most effective way for organizations to regain the initiative and become proactive, rather than reactive, with their IT security is to engage in threat modeling. Military strategists have used the concept of threat modeling for millennia. It is a means of analyzing one's security, assets, and capabilities from the attacker's perspective – allowing for the identification and prioritization of potential threats. Limited resources can then be applied to the most critical threats first, significantly enhancing the security posture without increasing the required resources.

Threat modeling came into the InfoSec mainstream in the early 2000s.[ii] The goal was to build security into applications at the design stage. Compared to the cost of remediating vulnerabilities discovered during scanning and pen-testing, initial secure coding is about 15x less expensive.[iii] Moreover, threat modeling reduces enterprise-wide exposure to application risk by identifying and recommending mitigating security controls for potential threats that vulnerability scanning and pen-testing miss.

Threat Modeling Tools

In response to the growing popularity of threat modeling, Microsoft developed a free tool, Microsoft SDL – first released in 2008 – to aid in the development of threat models. This tool was later replaced by Microsoft Threat Modeling Tool (TMT), which has an updated 2016 version. Microsoft's public domain tools were the only threat modeling tools widely available until ThreatModeler™ was first released in 2011.

The Microsoft tools are based on Microsoft's threat modeling methodology (sometimes referred to as the STRIDE methodology) – which is focused on promoting secure initial coding in

Microsoft's development environment for the Windows platform.[iv] This methodology also requires users to build threat models using data flow diagrams[v] – a throwback to the 1970s-era system engineering abstraction of how data is moved, stored, and manipulated by a single application. As a result, the Microsoft tools have limited functionality as an enterprise-level threat modeling tool.

ThreatModeler™, on the other hand, is based on the Visual, Agile, and Simple Threat modeling methodology (VAST).[vi] This methodology was specifically designed to support DevOps teams working within Agile methodologies and to allow an organization to scale its threat modeling practice across hundreds or even thousands of threat models without a significant increase in required resources. Creating an application threat model in ThreatModeler™ begins with the creation of a visual representation of the application using a process flow diagram.[vii] Process flow diagrams represent applications in the same way application architects and developers whiteboard an application during the design phase This allows developers or other stakeholders without specific security expertise can create, update, and interpret the visual decompositions of the applications for which they are creating threat models.

Furthermore, well beyond the capabilities of TMT, ThreatModeler™ also supports creation of operational threat models.[viii] Operational threat models allow the operations teams to create an end-to-end threat model of the organizations entire IT infrastructure system.

Moreover, with ThreatModeler™, individual threat models can be chained together, or nested one within another.[ix] This allows organizations to identify and contextually prioritize the mitigating strategies for potential threats inherent to application interactions, shared infrastructure components, and 3rd party elements.

Features Comparison

Recently, members of the security community have requested a comparison between ThreatModeler™ and Microsoft's TMT. In response, and in collaboration with independent sources, I created the following matrix to provide a head-to-head comparison:

	Microsoft TMT	ThreatModeler
Threat Library		
Built-in Threat Library Contains a pre-made common threat library developed from industry standards and security best practices	✔ 70 Windows-centric threats	✔ 600+ Web, Mobile, Cloud, & Infrastructure threats
Customizable Threat Library Allows users to add threats based on organizational internal research or commercial threat intelligence	✔	✔
Prioritization of Contextual Risks Provides a means for users to customize threat profiles to match with existing security policy	✔	✔

	Microsoft TMT	ThreatModeler
Threat Library *(continued)*		
Threat Library Updates Provides updates to the threat libraries with real-world threat intelligence	**Limited** Annual or longer	**Quarterly**
API to Upload Existing Threats Allows users to add existing threat intelligence from other commercial or internal sources	✗	✓
Prioritization of Organizational Risks (High Priority) Allows users to define high priority organization risks and relate those risks to real-world threat intelligence	✗	✓
Mapping Threats to Security Controls Provides users the ability to custom-define security controls and automatically correlate the controls to specific threats	✗	✓
Creating Useful Threat Models		
Time & Resources to Build a Threat Model* Average time for one personnel resource to create an application threat model for a median-sized application	**100-120 hours**	**16-24 hours**
Actionable Output Creates specific, actionable output for all SDLC stakeholders	**Limited**	✓
Reusability & Repeatability Allows users to reuse or embed threat model components from other threat models; enable users to chain individual application threat models into an end-to-end operational threat model	✗	✓
Advanced Modeling		
Ability to create Templates Allows users to customize threat model components as required for organizational need and application architecture	**Limited**	✓
Supports Attack Modeling Users can view an application or infrastructure from an attacker's perspective, including attack trees and attack surface analysis	✗	✓
Threat Model Chaining Users may link threat models together, providing the capacity to understand complex relationships resulting from sharing infrastructure components and application interactions, and the threats inherent to the resultant interconnectivity	✗	✓
Threat Tracability (Including Mitigation & Verification) Provides users with the capacity to trace an individual threat from the attack surface to its origin in each affected application	✗	✓

	Microsoft TMT	ThreatModeler
Reporting		
Threat Management Dashboard Automated dashboard providing the current, real-time status of identified threats at a glance	✔	✔
Architecture Diagrams Automatically provides security requirement checklist for hardening each infrastructure component	✔	✔
Out of the Box Reports Tool comes with pre-defined reports for security teams and other stakeholders	**Limited** 1	✔ 5
Threat Comparison and Trend Analysis Allow users to view and compare trends across multiple releases of an application or across multiple applications	✘	✔
Security		
Role-Based Access Controls to Threat Models Allows for the assignment of different access and permissions based on individual stakeholder roles and responsibilities	✘	✔
Secure Coding Guidelines Automatically provides development teams with relevant secure coding to mitigate prioritized threats for each threat model component	✘	✔
Support Hotline Tool comes with technical support for operational or functional assistance	✘	✔
Organization-Wide Security Policy Enforcement Provides users with a centralized threat library which automatically links to all threat models and application components, allowing threat models to be simultaneously updated for new relevant threats	✘	✔
Software Platform		
Component Based Design Allows users to build threat models based on components such as web services, ports, database services, and communication protocols	✔	✔
Customizable Data Elements, Widgets, Protocols, and such Provides users the capacity for customizing threat model compoents based on the organization's application architecture	**Limited**	✔
System Updates Time frame to receive functionality and other system updates	**Annual** or longer	**Quarterly**

	Microsoft TMT	ThreatModeler
Software Platform *(continued)*		
Platform Independence Users can utilize the tool on different computing platforms	✗ Windows-based	✔ Web-based
Collaborative Environment Allows multiple stakeholders to access the tool simultaneously and make real-time changes or updates	✗ Stand-alone	✔
Enterprise-Level Scalability Provides the capacity to build and maintain hundreds or even thousands of threat models that reside on different infrastructure stacks enterprise-wide	✗	✔
Integration with Existing Tools & Technologies The software has the functionality to provide bi-directional integration with other tools, technologies, and applications	✗	✔
Technical Support Tool comes with access to operational or functional support	✗	✔

* Time and resources needed to build a threat model for a medium sized application were supplied by an independent source, who documented the time spent by using both products to build a threat model for the exact same application.

Conclusion

Even though ThreatModeler™ requires an initial investment and an ongoing subscription, it provides organizations with far more features and capabilities than Microsoft's Threat Modeler Too. These additional features and capabilities innately enhance the organization's threat modeling capacity and provide the outputs organizations need to understand their real-time risk profile, the most important threats faced by the organization, and the organization's comprehensive attack surface.

Using the "free" Microsoft TMT will cost organizations significantly more in terms of ongoing labor, missed opportunities, and lack of necessary information to reduce risk organization-wide.

---

[i] "Cyber Security Market worth 202.36 Billion USD by 2021." MarketsandMarkets.com. 2016 http://www.marketsandmarkets.com/PressReleases/cyber-security.asp
.

[ii] "Threat Modeling 101." ThreatModeler.com. 2016. http://threatmodeler.com/threat-modeling-101/

[iii] Tassey, Gregory. "The Economic Impacts of Inadequate Infrastructure for Software Testing." RTI Health, Social, and Economics Research. National Institute of Standards and Technology: Gaithersburg, MD. May, 2002. https://www.nist.gov/sites/default/files/documents/director/planning/report02-3.pdf

[iv] "Threat Model." Wikipedia.com. https://en.wikipedia.org/wiki/Threat_model

[v] Agarwal, Archie. “Threat Modeling – Data Flow Diagram vs Process Flow Diagram.” ThreatModeler.com. August 18 2016. http://threatmodeler.com/threat-modeling-data-flow-diagram-vs-process-flow-diagram/

[vi] “Threat Modeling Methodology.” ThreatModeler.com. 2016. http://threatmodeler.com/threat-modeling-methodology/

[vii] Agarwal, Archie. “Threat Modeling – Data Flow Diagram vs Process Flow Diagram.” ThreatModeler.com. August 18 2016. http://threatmodeler.com/threat-modeling-data-flow-diagram-vs-process-flow-diagram/

[viii] Agarwal, Archie. “Application Threat Modeling vs Operational Threat Modeling.” ThreatModeler.com. September 6, 2016. http://threatmodeler.com/application-threat-modeling-vs-operational-threat-modeling/

[ix] “Threat Model Chaining.” ThreatModeler.com. 2016. http://threatmodeler.com/threat-model-chaining/