SRC CYBER SOLUTIONS LLP

THREATX

TECHNICAL BRIEF

Understanding ThreatX Detections

ThreatX detects a wide range of threats and malicious behaviors that can put web and API applications at risk, including but not limited to OWASP Top 10 threats (injection attacks, Layer 7 and bot-based attacks) as well as DDoS attacks. To support this broad spectrum of attack types, ThreatX combines a variety of detection strategies:

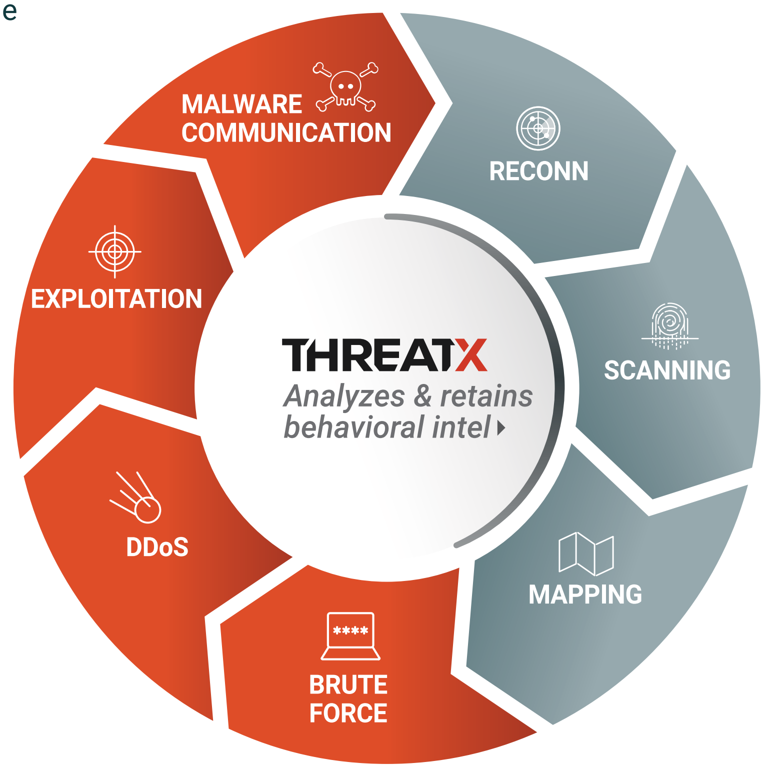
1. **Application profiling**
2. **Attacker profiling and monitoring**
3. **Active attacker interrogation**

All these detection methods are correlated across the lifecycle of an attack to provide a complete view of the risk to the application.

Detection States and Threat Classification Types

ThreatX detects and follows threats throughout the entire attack lifecycle. The phases of attacks are considered "states" and include the following:

- » Reconnaissance
- » Scanning
- » Mapping
- » Brute Force
- » DDos
- » Exploitation
- » Malware Communication



Attack tactics are grouped into "classifications". These classifications include categories like, but not limited to:

- » SQL Injection
- » Software Detection
- » Botnet Actvity
- » Evasion
- » Directory Traversal

It is important to note that some attack tactic classifications can apply to multiple attack states. For example, evasion techniques can be used during Reconnaissance, Web Mapping, as well as Exploitation.

continued on next page

The table below summarizes the various detection states and provides a few examples of typical threat classifications found in each.

ATTACK STATES	ATTACK TACTIC CLASSIFCATIONS
RECONNAISSANCE **The earliest phase of an attack: attempt to discover the basic building blocks of an application**	An attacker's first step may be attempt to determine how a site is built, such as PHP or WordPress (an example of the Software Detection classification). Other important classifications include Info Disclosure, Content Enumeration, Programmatic Access (e.g. attempting to use wget, curl), Evasion (e.g. attackers hiding behind Tor), BadBots (e.g. crawlers), and ToolKits.
SCANNING **A highly automated phase of attack that picks up where Reconn leaves off.**	This phase includes a wide variety of ThreatX classifications such as Content Enumeration (e.g. Drupal page enumeration), Plugin Enumeration (e.g. Easy WP SMTP), Username Enumeration, and Software Detection. The reliance on automated scanning also allows ThreatX to detect a variety of automated technologies with classifications for Botnet Activity, Toolkits (e.g. nmap, nikto), and Badbots.
WEB MAPPING **Attackers attempt to dig deeper into the application in order to discover paths or parts of the application that may not have been initially visible.**	Directory Traversal is one of the most important classifications during this phase and covers a wide variety of techniques attackers use as they attempt to move to protected areas of an application. Detection of Evasion is critically important during this phase as attackers will use encoding to hide their traversal attempts. Other relevant classifications include Password Guessing and SQL Injection such as x path injection.
BRUTE FORCE **Typified by attackers trying to break into accounts (user or administrative) even without having valid credentials.**	Error Rates classifications such as tracking login failures can be used to detect basic brute force attacks. ThreatX also includes more sophisticated models used to specifically detect techniques such as Password Spraying, and Credential Stuffing , which is commonly used by bots as part of account take-over attacks (ATO).
DDoS **Used by attackers in order to make an application or service inaccessible by valid users.**	DDoS attacks can take a variety of forms including volumetric attacks detected by the Traffic Flood classification. However, more sophisticated DDoS attacks have shifted to Layer 7 attacks which are detected by application profiling, bot detection, or detections designed to find Business Logic Abuse.
EXPLOITATION **Focuses on direct attacks against the application and a wide variety of threat classifications apply.**	This includes many of the traditional detections of a WAF including Command Injection, SQL Injection, XSS, Session Hijacking, and Remote File Inclusion.ThreatX also looks for exploitation of Known Vulnerability, Botnet Activity such as shell access, Form Spam, Business Logic Abuse,and a variety of Evasion techniques.
MALWARE COMMUNICATIONS **Refers to the tell-tale traffic created by malware as part of an ongoing, coordinated attack.**	These detections are often related to a post-exploitation phase of attack in which the attacker's malware has achieved persistence. Key classifications categories include Botnet Activity and Trojan Activity.

Attack and Threat Contexts

ThreatX compliments detection states and classifications with a variety of contexts to reveal suspicious activity that may indicate an attack, or provide additional context to confirm a threat or raise risk levels:

Application Profiling

ThreatX automatically learns the normal behavior of applications, services, and underlying application technologies such as Drupal, Java, RubyonRails, JSON API, SOAP API, and others. Deviations and anomalies can often provide an early indicator of a threat that may not match any known patterns or signatures. The ThreatX SOC can quickly identify common traits in anomalous traffic to create custom countermeasures that stop threats without impacting valid users.

Attacker Fingerprinting and Profiling

ThreatX actively fingerprints attackers and their attacking infrastructure to follow the attacker over time and across locations. This connects a coordinated attack that may span multiple phases (states) and may involve a variety of origins.

Attacker Interrogation

ThreatX actively challenges suspicious entities to reveal important information about the attacker. For example, ThreatX may reveal a bot or automation framework masquerading as a user by testing to see how the entity responds to cookies or javascript challenges. These techniques are completely transparent to a valid end user but can be very powerful tools to reveal the true nature of a bot.