SALT

Mapping the MITRE ATT&CK Framework to API Security

WHITE PAPER

The MITRE ATT&CK framework has been used for years by security professionals. This comprehensive matrix identifies, categorizes, and describes tactics, techniques, and procedures (TTPs) used by attackers. While no specific ATT&CK matrix for API security exists today, bad actors apply many of the tactics and techniques identified in the ATT&CK framework during the lifecycle of their API attack campaigns.

Understanding how these tactics and techniques relate to API security and how to apply that understanding can help organizations advance in their API security maturity.

What is the MITRE ATT&CK Framework

The MITRE ATT&CK Framework provides a knowledge base of information on cyber adversaries, their tactics, and how to defend against them. Organized by attack phases, or tactics, the framework lists the specific techniques that adversaries may use throughout an attack lifecycle within each tactic. While MITRE has developed different ATT&CK matrices focused on specific platforms and environments, no matrix for API security exists today. This content focuses on the Enterprise Matrix, version 12 (April 2022), which is a superset of all of the matrices. The tactics included in the Enterprise Matrix include:

1. Reconnaissance: Techniques attackers use to gather information about the target organization and its systems
2. Resource Development: Techniques attackers use to prepare an attack, such as creating payloads and scripts or weaponizing vulnerabilities
3. Initial Access: The techniques attackers use to gain their initial foothold in the target environment
4. Execution: The techniques attackers use to run their malicious code on the target systems
5. Persistence: The techniques attackers use to maintain their presence on the target systems
6. Privilege Escalation: The techniques attackers use to gain higher levels of access to the target systems
7. Defense Evasion: The techniques attackers use to evade detection and analysis
8. Credential Access: The techniques attackers use to gain access to valid credentials

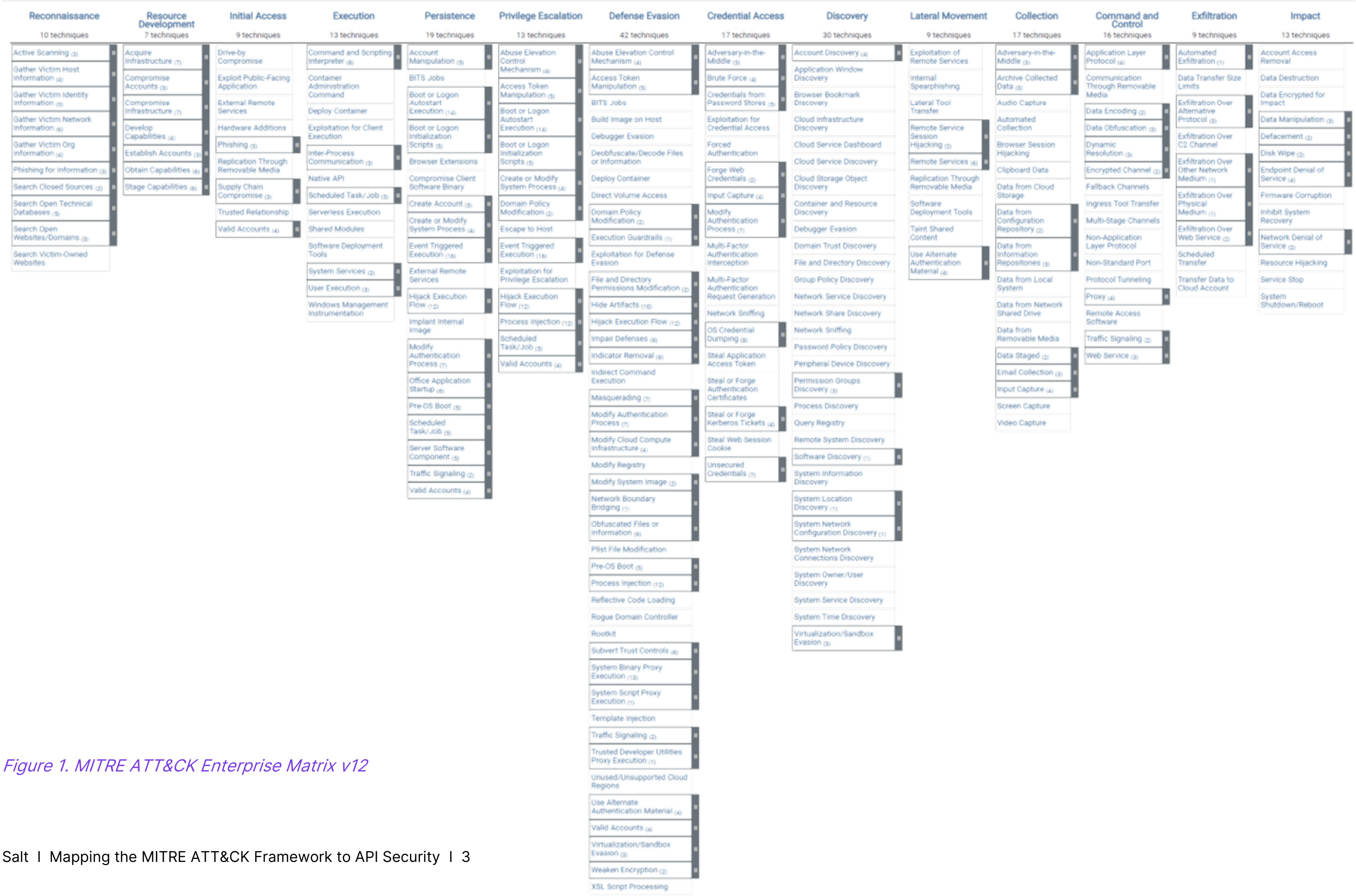
9. Discovery: The techniques attackers use to gather information about the target environment
10. Lateral Movement: The techniques attackers use to move through the target environment and access additional systems
11. Collection: The techniques attackers use to gather sensitive data from the target environment
12. Command and Control: The techniques attackers use to communicate with and control their malware or malicious code on the target systems
13. Exfiltration: The techniques attackers use to remove data from the target environment
14. Impact: The techniques attackers use to disrupt or destroy the target environment

First developed in 2013 by the cybersecurity experts at MITRE Corporation, the MITRE ATT&CK Framework originally focused on threats used against Windows enterprise networks. The goal of the comprehensive database was to allow those working within information security to better keep up with changing tactics employed by malicious actors and stay ahead of cyber threats.

The framework has grown exponentially in terms of use and recognition since its release. However, it would not be where it is today without the efforts of its many contributors. Presently, hundreds of contributors actively work on maintaining and regularly updating this open source knowledge base.

Today, cybersecurity professionals use this framework to identify potential threats posed by malicious actors targeting their networks or systems. By understanding the various TTPs identified in the MITRE ATT&CK Framework related to their specific environment, organizations can take steps to protect themselves from them.

For more detailed information on MITRE ATT&CK, its history, usage, and details on each of the TTPs associated with it, please visit https://attack.MITRE.org.



Reconnaissance
10 techniques

- Active Scanning (3)
- Gather Victim Host Information (4)
- Gather Victim Identity Information (3)
- Gather Victim Network Information (6)
- Gather Victim Org Information (4)
- Phishing for Information (3)
- Search Closed Sources (2)
- Search Open Technical Databases (5)
- Search Open Websites/Domains (3)
- Search Victim-Owned Websites

Resource Development
7 techniques

- Acquire Infrastructure (7)
- Compromise Accounts (3)
- Compromise Infrastructure (7)
- Develop Capabilities (4)
- Establish Accounts (3)
- Obtain Capabilities (6)
- Stage Capabilities (6)

Initial Access
9 techniques

- Drive-by Compromise
- Exploit Public-Facing Application
- External Remote Services
- Hardware Additions
- Phishing (3)
- Replication Through Removable Media
- Supply Chain Compromise (3)
- Trusted Relationship
- Valid Accounts (4)

Execution
13 techniques

- Command and Scripting Interpreter (8)
- Container Administration Command
- Deploy Container
- Exploitation for Client Execution
- Inter-Process Communication (3)
- Native API
- Scheduled Task/Job (5)
- Serverless Execution
- Shared Modules
- Software Deployment Tools
- System Services (2)
- User Execution (3)
- Windows Management Instrumentation

Persistence
19 techniques

- Account Manipulation (5)
- BITS Jobs
- Boot or Logon Autostart Execution (14)
- Boot or Logon Initialization Scripts (5)
- Browser Extensions
- Compromise Client Software Binary
- Create Account (3)
- Create or Modify System Process (4)
- Event Triggered Execution (16)
- External Remote Services
- Hijack Execution Flow (12)
- Implant Internal Image
- Modify Authentication Process (7)
- Office Application Startup (6)
- Pre-OS Boot (5)
- Scheduled Task/Job (5)
- Server Software Component (5)
- Traffic Signaling (2)
- Valid Accounts (4)

Privilege Escalation
13 techniques

- Abuse Elevation Control Mechanism (4)
- Access Token Manipulation (5)
- Boot or Logon Autostart Execution (14)
- Boot or Logon Initialization Scripts (5)
- Create or Modify System Process (4)
- Domain Policy Modification (2)
- Escape to Host
- Event Triggered Execution (16)
- Exploitation for Privilege Escalation
- Hijack Execution Flow (12)
- Process Injection (12)
- Scheduled Task/Job (5)
- Valid Accounts (4)

Defense Evasion
42 techniques

- Abuse Elevation Control Mechanism (4)
- Access Token Manipulation (5)
- BITS Jobs
- Build Image on Host
- Debugger Evasion
- Deobfuscate/Decode Files or Information
- Deploy Container
- Direct Volume Access
- Domain Policy Modification (2)
- Execution Guardrails (1)
- Exploitation for Defense Evasion
- File and Directory Permissions Modification (2)
- Hide Artifacts (10)
- Hijack Execution Flow (12)
- Impair Defenses (9)
- Indicator Removal (9)
- Indirect Command Execution
- Masquerading (7)
- Modify Authentication Process (7)
- Modify Cloud Compute Infrastructure (4)
- Modify Registry
- Modify System Image (2)
- Network Boundary Bridging (1)
- Obfuscated Files or Information (9)
- Plist File Modification
- Pre-OS Boot (5)
- Process Injection (12)
- Reflective Code Loading
- Rogue Domain Controller
- Rootkit
- Subvert Trust Controls (6)
- System Binary Proxy Execution (13)
- System Script Proxy Execution (1)
- Template Injection
- Traffic Signaling (2)
- Trusted Developer Utilities Proxy Execution (1)
- Unused/Unsupported Cloud Regions
- Use Alternate Authentication Material (4)
- Valid Accounts (4)
- Virtualization/Sandbox Evasion (3)
- Weaken Encryption (2)
- XSL Script Processing

Credential Access
17 techniques

- Adversary-in-the-Middle (3)
- Brute Force (4)
- Credentials from Password Stores (5)
- Exploitation for Credential Access
- Forced Authentication
- Forge Web Credentials (2)
- Input Capture (4)
- Modify Authentication Process (7)
- Multi-Factor Authentication Interception
- Multi-Factor Authentication Request Generation
- Network Sniffing
- OS Credential Dumping (8)
- Steal Application Access Token
- Steal or Forge Authentication Certificates
- Steal or Forge Kerberos Tickets (4)
- Steal Web Session Cookie
- Unsecured Credentials (7)

Discovery
30 techniques

- Account Discovery (4)
- Application Window Discovery
- Browser Bookmark Discovery
- Cloud Infrastructure Discovery
- Cloud Service Dashboard
- Cloud Service Discovery
- Cloud Storage Object Discovery
- Container and Resource Discovery
- Debugger Evasion
- Domain Trust Discovery
- File and Directory Discovery
- Group Policy Discovery
- Network Service Discovery
- Network Share Discovery
- Network Sniffing
- Password Policy Discovery
- Peripheral Device Discovery
- Permission Groups Discovery (3)
- Process Discovery
- Query Registry
- Remote System Discovery
- Software Discovery (1)
- System Information Discovery
- System Location Discovery (1)
- System Network Configuration Discovery (1)
- System Network Connections Discovery
- System Owner/User Discovery
- System Service Discovery
- System Time Discovery
- Virtualization/Sandbox Evasion (3)

Lateral Movement
9 techniques

- Exploitation of Remote Services
- Internal Spearphishing
- Lateral Tool Transfer
- Remote Service Session Hijacking (2)
- Remote Services (6)
- Replication Through Removable Media
- Software Deployment Tools
- Taint Shared Content
- Use Alternate Authentication Material (4)

Collection
17 techniques

- Adversary-in-the-Middle (3)
- Archive Collected Data (3)
- Audio Capture
- Automated Collection
- Browser Session Hijacking
- Clipboard Data
- Data from Cloud Storage
- Data from Configuration Repository (2)
- Data from Information Repositories (3)
- Data from Local System
- Data from Network Shared Drive
- Data from Removable Media
- Data Staged (2)
- Email Collection (3)
- Input Capture (4)
- Screen Capture
- Video Capture

Command and Control
16 techniques

- Application Layer Protocol (4)
- Communication Through Removable Media
- Data Encoding (2)
- Data Obfuscation (3)
- Dynamic Resolution (3)
- Encrypted Channel (2)
- Fallback Channels
- Ingress Tool Transfer
- Multi-Stage Channels
- Non-Application Layer Protocol
- Non-Standard Port
- Protocol Tunneling
- Proxy (4)
- Remote Access Software
- Traffic Signaling (2)
- Web Service (3)

Exfiltration
9 techniques

- Automated Exfiltration (1)
- Data Transfer Size Limits
- Exfiltration Over Alternative Protocol (3)
- Exfiltration Over C2 Channel
- Exfiltration Over Other Network Medium (1)
- Exfiltration Over Physical Medium (1)
- Exfiltration Over Web Service (2)
- Scheduled Transfer
- Transfer Data to Cloud Account

Impact
13 techniques

- Account Access Removal
- Data Destruction
- Data Encrypted for Impact
- Data Manipulation (3)
- Defacement (2)
- Disk Wipe (2)
- Endpoint Denial of Service (4)
- Firmware Corruption
- Inhibit System Recovery
- Network Denial of Service (2)
- Resource Hijacking
- Service Stop
- System Shutdown/Reboot

Figure 1. MITRE ATT&CK Enterprise Matrix v12

The Relationship between MITRE ATT&CK and API Security

The MITRE ATT&CK framework draws on real-world observations to provide a comprehensive understanding of the different tactics and techniques that adversaries may use in cyber attacks. Such a broad knowledge base cannot neatly map to a specific cyber area such as API security. Despite not having an API security-specific matrix today, many of the TTPs identified in the MITRE framework are also used by API attackers as techniques used in the different phases of their attack campaigns.

For example, an attacker may use reconnaissance techniques such as scanning public APIs or enumerating endpoints to identify potential targets. They might also attempt privilege escalation or abuse API tokens to gain access to sensitive information or resources. All of these behaviors fit within various techniques of the MITRE ATT&CK framework under the Reconnaissance, Discovery and Privilege Escalation tactics.

Benefits of Understanding the MITRE ATT&CK/API Security Relationship

With API security top of mind for CISOs and cybersecurity professionals worldwide, it's not surprising that they want solutions that can map attacker behavior directly back to the MITRE ATT&CK Framework. By understanding how attackers target APIs, organizations can quickly identify threats and take action accordingly. In addition, they can develop more effective defense strategies that align with their existing processes, procedures, policies, and technologies.

More specifically, understanding the relationship between the MITRE ATT&CK framework and API security can provide several benefits to security organizations, including:

1. Improved threat detection: By understanding the specific TTPs used by attackers against APIs, organizations can better detect and respond to threats targeting their APIs.
2. More effective incident response: The ATT&CK framework can help incident responders quickly identify the stage of an attack, the TTPs used, and take appropriate action.
3. Better resource allocation: By understanding the most likely attack methods and the resources required to defend against them, organizations can make more informed decisions about where to allocate security resources.

4. Improved communication: The ATT&CK framework provides a common language for discussing threats, making it easier for different teams within an organization to communicate about security issues.
5. Increased ability to measure and improve security: By basing security efforts on the ATT&CK framework, organizations can measure their progress against specific attack methods and continually improve their defenses.
6. Better understanding of the scope of an attack: The ATT&CK framework provides a comprehensive view of the different TTPs that can be used in an attack and the different stages of the attack, which can help organizations to better understand the scope of the attack and the potential impact.
7. Help identify security gaps: By identifying the TTPs relevant for specific APIs, organizations can identify the gaps in their security approach and take the necessary measures to mitigate the risks.

The Relationship Between MITRE ATT&CK and the OWASP API Security Top 10

When organizations discuss API security, they commonly reference API security threats in terms of the OWASP API Security Top 10. The OWASP API Security Top 10 list outlines the most critical security risks for web APIs, published by the Open Web Application Security Project (OWASP). Originally published in 2019, a new updated top 10 list for APIs is expected in 2023. To learn about the OWASP API Security Top 10 in greater detail, click here.

The OWASP API Security Top 10 focuses on the security risks specific to web APIs, while the MITRE ATT&CK framework provides a more general understanding of the TTPs used by adversaries in cyber attacks, including attacks on APIs. While the two frameworks address different aspects of security, they share a direct relationship.

Consider an API attack where the attacker is looking to exfiltrate data out of an API through a BOLA (Broken Object Level Authorization) vulnerability (the top threat on the OWASP API Top 10 list). The attacker would typically go through a series of phases, each with specific behaviors and exercises, to construct and carry out that attack campaign. Many of those behaviors map directly to MITRE ATT&CK tactics and techniques (TTPs). Think of TTPs as the ingredients an attacker would use when they are cooking up their BOLA API attack recipe.

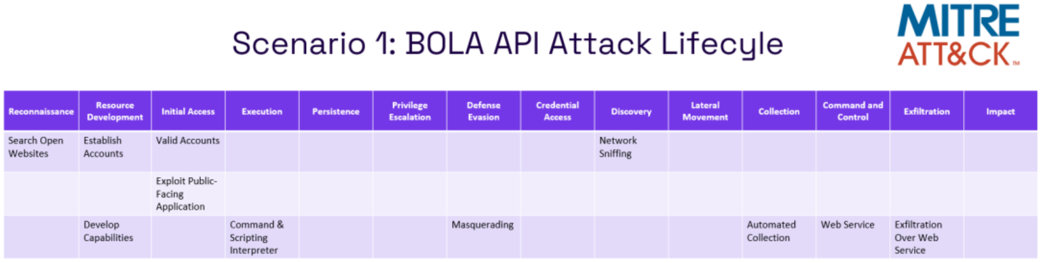
Moreover, organizations should know that a single OWASP API Security Top 10 threat does not map to a single set of MITRE ATT&CK TTPs. Remember, TTPs describe the phases, tactics and techniques of an attacker. When attacking a BOLA vulnerability (OWASP API 1), an attack can take many execution paths, leveraging various MITRE ATT&CK tactics and techniques. Actual attack sequences may differ and use different techniques depending on the specific situation and the capabilities of the attacker, as well as the specifics of the vulnerability and the implementation of the API. In that regard, organizations must remember that the relationship between an OWASP API Security Top 10 threat and MITRE ATTACK TTPs is a one-to-many relationship.

Mapping API Attack Scenarios to MITRE ATT&CK TTPs

Because each API is different in its purpose, function, and method of deployment, and because each attacker is different in terms of capabilities, no one can map every single API attack scenario permutation to MITRE ATT&CK. In this discussion, we aim to illustrate how organizations can better relate the behaviors of API attackers back to the TTPs documented in MITRE ATT&CK. Below, we outline a few different and common API attack scenarios that echo recent API breaches drawn from the headlines and show how the behaviors of the attacker map to MITRE ATT&CK TTPs.

API Attack Scenario 1: BOLA

Scenario: Attacker uses his/her own legitimate API credentials to exfiltrate other customer data through a BOLA vulnerability in the API discovered by the attacker. Examples of breaches in the headlines related to this scenario include Facebook, Experian, Expedia, Coinbase.



Scenario 1: BOLA API Attack Lifecyle

MITRE ATT&CK

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Search Open Websites	Establish Accounts	Valid Accounts						Network Sniffing					
		Exploit Public-Facing Application											
	Develop Capabilities		Command & Scripting Interpreter			Masquerading				Automated Collection	Web Service	Exfiltration Over Web Service	

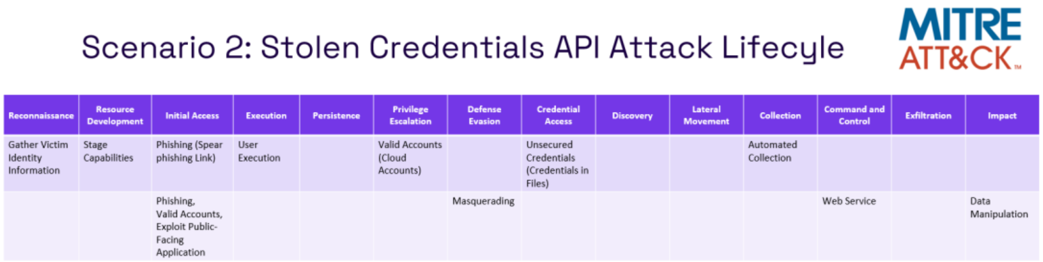
Figure 2: BOLA API Attack Lifecycle mapped to MITRE ATT&CK Framework

Mapping (Tactic/Technique):

1. Reconnaissance / Search Open Websites: Adversary researches information about a particular company service or application.
2. Resource Development / Establish Accounts: Adversary creates temporary email and mobile accounts to use in attack.
3. Initial Access / Valid Accounts: Adversary obtains valid application credentials on the target application through legitimate means, such as registering for a new account with the targeted service.
4. Discovery / Network Sniffing: Adversary examines the traffic passed from browser to the API to determine the available API functionality and data.
5. Initial Access / Exploit Public-Facing Application: Adversary takes learnings from the Discovery / Network Sniffing exercise and actively attempts to find logic flaws or vulnerabilities in the API that can be exploited. Adversary finds an API endpoint that is susceptible to a BOLA attack.
6. Resource Development / Develop Capabilities: Adversary develops a script to assist in automating the attack and exfiltrating user data through the vulnerable API endpoint.
7. Execution / Command and Scripting Interpreter: Adversary runs attack script on local or remote systems to actively exploit BOLA vulnerability.
8. Defense Evasion / Masquerading: Adversary manipulates data payloads and rate of API requests to evade rate limits and other security detection techniques provided by devices such as a WAF.
9. Collection / Automated Collection: Adversary uses the script written specifically to access the API and extract sensitive data, such as user information or financial data.
10. Command and Control / Web Service: Adversary maintains a command and control channel through the API to maintain access to it, exercise it, and continue to exfiltrate data through it over time.
11. Exfiltration / Exfiltration Over Web Service: Adversary uses the API to exfiltrate the stolen data out of the targeted organization's network.Overall, understanding the relationship between the MITRE ATT&CK framework and API security can help organizations to better protect their APIs, accelerate their response to attacks, and improve their overall API security posture.

API Attack Scenario 2: Stolen Credentials

Scenario: Attacker obtains API credentials through nefarious means – we'll use spear phishing as the example here – and leverages those credentials to exercise a privileged API in a malicious manner and compromise the integrity of the service it provides. This scenario shows how two attacks can be chained together as part of a whole API attack lifecycle. In this scenario, the attacker first gains access to the target company's online source code repository through social engineering to obtain privileged API credentials, then abuses the API to compromise service integrity. Examples of breaches in the headlines related to this scenario include CircleCI, Dropbox, and Slack.


Scenario 2: Stolen Credentials API Attack Lifecyle

MITRE ATT&CK™

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Gather Victim Identity Information	Stage Capabilities	Phishing (Spear phishing Link)	User Execution		Valid Accounts (Cloud Accounts)		Unsecured Credentials (Credentials in Files)			Automated Collection			
		Phishing, Valid Accounts, Exploit Public-Facing Application				Masquerading					Web Service		Data Manipulation

Figure 3: Stolen Credentials API Attack Lifecycle mapped to MITRE ATT&CK Framework

Mapping (Tactic/Technique):

Spear Phishing Campaign

1. Reconnaissance / Gather Victim Identity Information: Adversary researches information about an individual who holds a particular technical job function, that would have access to the source control service the adversary is interested in attacking.
2. Resource Development / Stage Capabilities: Adversary stages malicious web application in a cloud service to mimic a popular source code repository service.
3. Initial Access / Phishing (Spear Phishing Link): Adversary sends spear phishing email with a security notice, masquerading as the source control service, informing the target that his/her account may have been compromised and provides a malicious link asking the individual to confirm the service password and set a new password.
4. Execution / User Execution: User clicks link and enters existing password into staged web application.
5. Privilege Escalation / Valid Accounts: Cloud Accounts: Adversary leverages the stolen, valid source code repository to gain access to the repository.

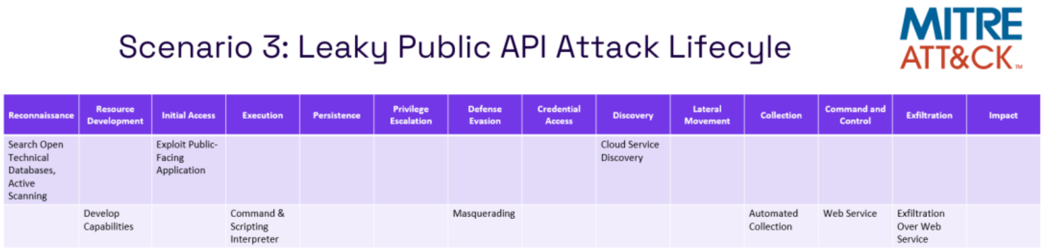
6. Collection / Automated Collection: Adversary leverages the stolen, valid source code repository to pull down copies of source code repositories.
7. Credential Access / Unsecured Credentials: Credentials in Files: Adversary scans source code repository for unencrypted/unsecured privileged API keys. Various keys and API addresses are captured.

API Misuse Campaign

1. Initial Access / Phishing: Adversary obtains valid API credentials through spear phishing attack as outlined above.
2. Initial Access / Valid Accounts: Adversary leverages the stolen, valid API credentials to access the API.
3. Initial Access / Exploit Public-Facing Application: Through trial and error, the adversary actively attempts to understand functions and capabilities of the API.
4. Defense Evasion / Masquerading: Adversary manipulates data payloads and rate of API requests to evade rate limits and other security detection techniques provided by devices such as a WAF.
5. Command and Control / Web Service: Adversary maintains a command and control channel through the API to maintain access to it, exercise it, and abuse the API.
6. Impact / Data Manipulation: Adversary manipulates data and service configurations.

API Attack Scenario 3: Leaky Public API

Scenario: Attacker identifies a publicly exposed API with a weak security posture (no authentication) that the attacker can leverage to exfiltrate customer data. Examples of breaches in the headlines related to this scenario include Twitter, LinkedIn, Peloton, T-Mobile, and Optus.



Scenario 3: Leaky Public API Attack Lifecyle

MITRE
ATT&CK™

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Search Open Technical Databases, Active Scanning		Exploit Public-Facing Application						Cloud Service Discovery					
	Develop Capabilities		Command & Scripting Interpreter			Masquerading				Automated Collection	Web Service	Exfiltration Over Web Service	

Figure 4: Leaky Public API Attack Lifecycle mapped to MITRE ATT&CK Framework

1. Reconnaissance / Search Open Technical Databases: Adversary researches domains owned by the targeted organization in various web and DNS registries and online databases.
2. Reconnaissance / Active Scanning: Adversary scans the discovered domains and IP ranges of the target company, specifically looking for any exposed APIs with weak security postures that can easily be taken advantage of, such as unauthenticated API endpoints.
3. Initial Access / Exploit Public-Facing Application: Upon finding an unauthenticated, publicly exposed API, the adversary begins to actively learn the function, purpose, and capabilities of the API and quickly learns the API is capable of returning sensitive customer data.
4. Discovery / Cloud Service Discovery: Adversary looks to learn about and discover other API services and endpoints that are accessible. (While there is currently no API discovery technique in the ATT&CK framework, in this scenario, the Cloud service discovery seems appropriate.)
5. Resource Development / Develop Capabilities: Adversary develops a script to assist in automating the attack and exfiltrating customer data through the vulnerable API endpoint.
6. Execution / Command and Scripting Interpreter: Adversary runs attack script on local or remote systems to actively exploit the API.
7. Defense Evasion / Masquerading: Adversary manipulates data payloads and rate of API requests to evade rate limits and other security detection techniques provided by devices such as a WAF.
8. Collection / Automated Collection: Adversary uses the script written specifically to access the API and extract sensitive data, such as user information or financial data.
9. Command and Control / Web Service: Adversary maintains a command and control channel through the API to maintain access to it, exercise it, and continue to exfiltrate data through it over time.
10. Exfiltration / Exfiltration Over Web Service: Adversary uses the API to exfiltrate the stolen data out of the targeted organization's network.

Shortcomings of Relating the MITRE ATT&CK Framework to API Security Threats

As discussed earlier, no ATT&CK matrix specifically geared toward API threats exists today. As a result, the broader framework has shortcomings when it comes to detailing all the potential aspects of an API attack.

For example, relying on the current framework for in-depth insight into defending against API attacks would be a mistake since it would miss many of the threats at play. We hope the MITRE ATT&CK framework will evolve to include a new matrix or list the techniques specifically pertinent to API attacks, as has happened with cloud and SaaS environments. Such an update would provide organizations with a more comprehensive understanding of API security threats and help them better protect their systems and data.

However, despite these shortcomings, organizations can gain tremendous value in mapping the relationship of these TTPs to the behaviors of adversaries during API attacks. This mapping can provide organizations with a deeper understanding of the specific tactics and techniques that are being used by attackers in API-based attacks, helping to predict and prepare for the types of attacks that are most likely to occur. Also, by understanding the relationships between different TTPs and the behaviors of API adversaries, organizations can better respond to attacks by leveraging systems and controls they already have in place based on the MITRE ATT&CK framework.

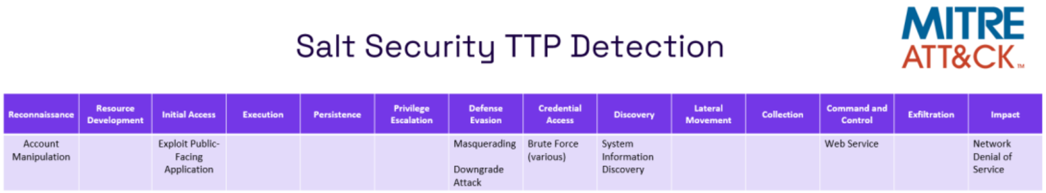
How Salt Security Leverages The MITRE ATT&CK Framework

The Salt Security API Protection Platform provides real-time protection against API-based attacks and uses the most advanced AI and machine learning techniques to continuously monitor API traffic and identify malicious behavior, even in the most complex and dynamic environments. The Salt platform not only maps real-time API threats to the OWASP API Security Top 10 but also supports the option to apply the MITRE ATT&CK framework to help identify and mitigate API-related threats.

Where applicable, the Salt platform maps identified threats to the relevant TTPs in the MITRE ATT&CK framework, providing organizations with a comprehensive view of the attack vectors and techniques being used by attackers. This information allows organizations to better understand the risks they face and prioritize their security efforts accordingly, as well as respond to live security threats using the

language, processes, and procedures already in place for various MITRE ATT&CK threats. Salt is committed to continuing to expand its support for the MITRE

ATT&CK framework over time as well as to work with the ATT&CK community to extend the framework's inclusion of API security-related threats.



Salt Security TTP Detection

Reconnaissance	Resource Development	Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Exfiltration	Impact
Account Manipulation		Exploit Public-Facing Application				Masquerading Downgrade Attack	Brute Force (various)	System Information Discovery			Web Service		Network Denial of Service

Figure 5: Salt Security Platform's Expanding TTP Detection

In Summary

By understanding how attacks against APIs relate to the MITRE ATT&CK framework, API-conscious security professionals can gain valuable insights into potential threat vectors that could be targeting their systems. They can also understand how best to defend against them – and they can do so in a security framework they are already accustomed to. This knowledge helps security teams more easily identify vulnerable areas of their API infrastructure, as well as develop more effective incident response plans should they experience a breach due to an attack on their APIs. In the future, the evolution of the MITRE ATT&CK framework to include more specific API security threats, behavior, and terminology will further extend and enhance its value.

WP-MITATT-02062023

SALT

Securing your Innovation.