From the Attacker's Perspective: List of Questions for Identifying Security Vulnerabilities in Your Web Application

@Bour Abdelhadi

Authentication and Authorization

- Can I easily guess or crack user passwords?
- Are passwords stored in plaintext or using weak encryption?
- Can I intercept or manipulate passwords in transit?
- Are password reset mechanisms secure? Can I reset a user's password without proper authorization?
- Can I bypass password requirements or restrictions?
- Can I hijack or steal a user's session token?
- Can I access another user's session data?
- Can I perform actions without proper authentication or authorization?
- Can I bypass session timeouts or invalidation?
- Can I manipulate session data to gain unauthorized access?
- Can I bypass multi-factor authentication requirements?
- Can I intercept or manipulate MFA tokens in transit?
- Can I bypass MFA requirements for sensitive actions?
- Can I use social engineering to bypass MFA requirements?
- Can I access sensitive data or functions without proper authorization?
- Can I bypass access controls or user roles?
- Can I manipulate user permissions or roles?
- Can I escalate my privileges or access level?
- Can I perform actions as another user?
- Can I bypass authorization requirements?
- Can I access sensitive resources or functions without proper authorization?
- Can I manipulate authorization mechanisms or tokens?
- Can I bypass OAuth, SAML, or other modern authorization mechanisms?
- Can I steal or manipulate token secrets?
- Can I reuse or manipulate tokens to gain unauthorized access?
- Can I intercept or manipulate tokens in transit?
- Can I bypass token invalidation or rotation?
- Can I perform brute-force attacks on user accounts?

- Can I bypass login protections, such as IP restrictions or password attempt limits?
- Can I manipulate or bypass account lockout measures?
- Can I use social engineering to gain unauthorized access?
- Can I bypass login input validation or security measures?

Input Validation

- Can I input malicious code (e.g. SQL injection, XSS, command injection) into the application?
- Can I bypass input validation or sanitization measures?
- Can I manipulate input data to perform unauthorized actions?
- Can I upload malicious files or other types of data?
- Are file upload paths properly restricted?
- Are input data length limits properly implemented?
- Are input fields vulnerable to buffer overflows or other memory exploits?
- Are input fields vulnerable to integer overflows or underflows?
- Are input fields vulnerable to regular expression denial-of-service (ReDoS) attacks?
- Are input fields vulnerable to LDAP injection attacks?
- Are input fields vulnerable to XML injection attacks?
- Are input fields vulnerable to XPath injection attacks?
- Are input fields vulnerable to XML external entity (XXE) attacks?
- Are input fields vulnerable to HTTP header injection attacks?
- Are input fields vulnerable to HTTP request smuggling attacks?
- Are input fields vulnerable to server-side request forgery (SSRF) attacks?
- Are input fields vulnerable to cross-site request forgery (CSRF) attacks?
- Are input fields vulnerable to local file inclusion (LFI) or remote file inclusion (RFI) attacks?
- Are input fields vulnerable to directory traversal attacks?
- Are input fields vulnerable to path traversal attacks?
- Are input fields vulnerable to JavaScript injection attacks?
- Can I upload malicious files to the application?
- Are file upload paths properly restricted?
- Are uploaded files properly scanned for malware?
- Are uploaded files being stored securely?
- Can I access or manipulate uploaded files after they have been stored?
- Are file extensions being properly validated?
- Are file size limits properly implemented?

- Are file content types being properly validated?
- Is client-side validation being used? Is it secure?
- Can I bypass client-side validation measures?
- Are client-side validation measures being enforced on the server side as well?
- Is client-side validation being used to prevent XSS attacks?
- Is client-side validation being used to prevent CSRF attacks?
- Is server-side validation being used? Is it secure?
- Can I bypass server-side validation measures?
- Are server-side validation measures being enforced consistently?
- Is server-side validation being used to prevent SQL injection or other attacks?
- Is server-side validation being used to prevent path traversal or directory traversal attacks?
- Is server-side validation being used to prevent LFI or RFI attacks?
- Is server-side validation being used to prevent XXE attacks?
- Is server-side validation being used to prevent CSRF attacks?
-

Data Storage and Encryption

- Can I access sensitive data stored in the application?
- Is sensitive data stored in plaintext or using weak encryption? Can I easily decrypt the data?
- Are there any backup or recovery mechanisms that could be exploited?
- Can I manipulate data stored in the application?
- Are database queries and access protected from SQL injection or other attacks?
- Are log files or audit trails available for me to exploit?
- Are data retention policies being properly implemented and enforced? Can I find sensitive data in old or outdated backups?
- Is encryption being used to protect sensitive data in transit and at rest? Can I bypass the encryption?
- Are encryption keys being stored securely? Can I steal or manipulate encryption keys?
- Are encryption algorithms and protocols up to date? Are there any known vulnerabilities?
- Is data being encrypted properly and consistently?
- Are there any weaknesses in the encryption mechanisms that can be exploited?
- Can I intercept encrypted data in transit?

API Security

- Can I access API endpoints without proper authentication?
- Are authentication mechanisms secure? Can I bypass authentication?
- Can I steal or manipulate access tokens to gain unauthorized access?
- Can I manipulate authentication headers or requests to bypass authentication?
- Are there any vulnerabilities in the API authentication flow that I can exploit?
- Are there any hardcoded API keys or secrets that I can use to bypass authentication?
- Are API endpoints being properly protected based on user roles and permissions?
- Can I access sensitive data or functions without proper authorization?
- Are there any vulnerabilities in the API authorization flow that I can exploit?
- Can I manipulate authorization mechanisms or tokens to gain unauthorized access?
- Can I bypass RBAC or other authorization mechanisms to gain unauthorized access?
- Can I input malicious code (e.g. SQL injection, XSS, command injection) into the API?
- Can I bypass input validation or sanitization measures?
- Can I manipulate input data to perform unauthorized actions?
- Can I upload malicious files or other types of data through the API?
- Are input data length limits properly implemented?
- Are input fields vulnerable to buffer overflows or other memory exploits?
- Are input fields vulnerable to integer overflows or underflows?
- Can I manipulate API responses to perform unauthorized actions?
- Are API responses being properly validated and sanitized?
- Are sensitive data or error messages being leaked through API responses?
- Are API responses being properly escaped or encoded?
- Are API responses being properly validated for data type and length?
- Is secure communication being used to protect API traffic?
- Are transport security mechanisms being properly implemented and configured?
- Can I intercept or manipulate API traffic in transit?
- Are there any vulnerabilities in the API transport security mechanisms that I can exploit?
- What web technologies are used in the application? Are there any known vulnerabilities or exploits associated with those technologies?
- Are there any public or exposed APIs that can be attacked or exploited?
- What type of data is being collected or processed by the application? Are there any sensitive or valuable data stores that can be targeted?

Logging and Monitoring

- Can I prevent or manipulate log generation to cover my tracks?
- Can I manipulate or delete logs to hide my actions or impact?
- Are sensitive data or credentials being stored in log files? Can I steal this information?
- Are log files being properly secured and protected from unauthorized access?
- Are log files being backed up and retained for an appropriate amount of time? Can I find sensitive data in old or outdated log files?
- Are log files being properly rotated to prevent file size issues or other potential problems?
- Are monitoring mechanisms in place to detect and alert on potential security incidents?
- Are alerts being generated for all important security events and actions?
- Are security events being triaged and investigated in a timely manner?
- Can I bypass monitoring mechanisms or generate false positives?
- Are monitoring mechanisms being tuned to reduce false positives and improve detection capabilities?
- Are monitoring mechanisms being audited and reviewed regularly to ensure they are effective in detecting and preventing attacks?

TPS

- Are third-party services being used in the application?
- Can I manipulate or bypass third-party service integrations?
- Are third-party services being used securely? Are they following security best practices?
- Are third-party services being used in compliance with data privacy and protection regulations?
- Are there any vulnerabilities in the third-party services that can be exploited?
- Are third-party services being properly monitored and audited for security vulnerabilities?
- Can I use third-party services to gain unauthorized access to the application or sensitive data?
- Are third-party libraries and dependencies being used in the application?
- Are third-party libraries and dependencies being kept up to date with security patches and updates?
- Are there any known vulnerabilities in the third-party libraries and dependencies that can be exploited?
- Are third-party libraries and dependencies being properly monitored and audited for security vulnerabilities?
- Can I use vulnerabilities in third-party libraries and dependencies to gain unauthorized

access to the application or sensitive data?

- Is the application supply chain secure? Can I manipulate or compromise the supply chain to introduce vulnerabilities or malicious code?
- Are there any vulnerabilities in the build or deployment process that can be exploited?
- Are there any untrusted or insecure components in the supply chain that can be exploited?
- Are there any known supply chain attacks or incidents that may impact the application?
- Are supply chain security measures being properly implemented and audited to prevent attacks?

Bonus

- Can I use HTTP smuggling attacks to bypass security mechanisms and gain unauthorized access to the application or sensitive data?
- Are there any unvalidated or unsanitized HTTP headers that could be used to exploit HTTP smuggling vulnerabilities?
- Can I use different HTTP protocols (e.g. HTTP/1.0, HTTP/1.1, HTTP/2) or methods (e.g. GET, POST, PUT, DELETE) to exploit HTTP smuggling vulnerabilities?
- Are there any discrepancies in the way that the application and backend servers interpret HTTP requests that can be exploited?
- Can I use content-length discrepancies or other HTTP header manipulation techniques to exploit HTTP smuggling vulnerabilities?
- Can I use chunked encoding or other HTTP body manipulation techniques to exploit HTTP smuggling vulnerabilities?
- Are there any reverse proxy or load balancer configurations that can be exploited to bypass HTTP smuggling protections?
- Can I use HTTP smuggling attacks to bypass rate limiting, IP blocking, or other security mechanisms?
- Can I exploit business logic flaws in the authentication or authorization mechanisms to gain unauthorized access?
- Can I use polymorphic payloads or evasion techniques to bypass input validation or sanitization measures?
- Are there any vulnerabilities in custom input validation or sanitization functions that can be exploited?
- Can I use alternate data encoding formats to bypass input validation or sanitization measures?
- Can I exploit session fixation vulnerabilities to hijack user sessions?

- Are there any vulnerabilities in the session management implementation that can be exploited?
- Can I use timing attacks or other side-channel attacks to exploit weaknesses in the encryption mechanisms?
- Are there any cryptographic key management vulnerabilities that can be exploited?
- Can I use chosen ciphertext attacks or other cryptanalysis techniques to bypass the encryption mechanisms?
- Are there any race conditions or other concurrency vulnerabilities that can be exploited?
- Can I use API or other interface vulnerabilities to bypass application logic or gain unauthorized access?
- Can I exploit vulnerabilities in third-party or open-source components used by the application?
- Are there any logical or functional vulnerabilities in the application that can be exploited?