**ÔN TẬP**

**TUẦN 7 - WAF Bypassing Techniques**

**Câu 1: "Chunked Transfer Encoding" có thể được sử dụng để bypass WAF như thế nào?**

**a. Chia nhỏ yêu cầu HTTP thành các phần để qua mặt WAF**

b. Gửi yêu cầu HTTP cùng với mã độc

c. Mã hóa toàn bộ yêu cầu

d. Loại bỏ các ký tự đặc biệt trong yêu cầu

**Câu 2: Ghép các WAF phổ biến với dấu hiệu nhận diện của chúng.**

**a. Cloudflare → Cookie chứa "\_cfduid", Imperva → Cookie chứa "incap\_ses\_" hoặc "visid\_incap\_", F5 Big IP → Cookie chứa chuỗi "TSxxxxxxxx"**

b. **Cloudflare** → Cookie chứa "\_cfduid", **F5 Big IP** → Cookie chứa "incap\_ses\_" hoặc "visid\_incap\_", **Imperva** → Cookie chứa chuỗi "TSxxxxxxxx"

c. **Imperva** → Cookie chứa "\_cfduid", **Cloudflare** → Cookie chứa "incap\_ses\_" hoặc "visid\_incap\_", **F5 Big IP** → Cookie chứa chuỗi "TSxxxxxxxx"

**Câu 3: Ghép các phương pháp kiểm tra với mô tả mục đích của chúng trong Command Injection.**

**a. Input Validation → Kiểm tra và xác minh đầu vào của người dùng, Output Encoding → Mã hóa đầu ra để tránh thực thi không mong muốn, Sanitization → Loại bỏ các ký tự không hợp lệ trong đầu vào**

b. **Input Validation** → Kiểm tra và xác minh đầu vào của người dùng, **Output Encryption** → Mã hóa đầu ra để tránh thực thi không mong muốn, **Sanitization** → Loại bỏ các ký tự không hợp lệ trong đầu vào

c. **Input Validation** → Kiểm tra và xác minh đầu vào của người dùng, **Output Encoding** → Mã hóa đầu ra để tránh thực thi không mong muốn, **Sanitization** → Loại bỏ các ký tự hợp lệ trong đầu vào

**Câu 4: Kỹ thuật "Data URI Encoding" có thể hỗ trợ bypass WAF trong trường hợp nào?**

**a. Gửi dữ liệu dưới dạng URI để qua mặt WAF**

b. Mã hóa dữ liệu đầu ra thành định dạng JSON

c. Loại bỏ các tham số trong URL

d. Thay đổi định dạng của IP Address

**Câu 5: Kỹ thuật nào dưới đây không phải là kỹ thuật bypass WAF?**

**a. Sử dụng giao thức HTTPS**

b. Chunked Request

c. Null Character Injection

d. Inline Comment

**Câu 6: Mục đích của kỹ thuật "Null Character Injection" trong bypass WAF là gì?**

**a. Đưa vào ký tự NULL để bypass kiểm tra chuỗi**

b. Mã hóa URL để tránh phát hiện

c. Thêm các tham số vào yêu cầu HTTP

d. Loại bỏ các khoảng trắng trong SQL Injection

**Câu 7: Mục đích của việc sử dụng "User-Agent Spoofing" là gì?**

**a. Giả mạo User-Agent để qua mặt WAF**

b. Chuyển đổi dữ liệu thành dạng JSON

c. Loại bỏ các ký tự nguy hiểm

d. Tăng tốc độ xử lý của HTTP request

**Câu 8: HTTP Parameter Pollution thường được sử dụng để hỗ trợ loại tấn công nào?**

**a. SQL Injection**

b. XSS

c. CSRF

d. File Upload

**Câu 9: Ghép các kỹ thuật bypass với các tình huống mà chúng thường được sử dụng.**

**a. Data URI Encoding → Bypass kiểm tra tệp tải lên, JSON Wrapping → Bypass kiểm tra dữ liệu không phải JSON, IP Spoofing → Giả mạo địa chỉ IP để tránh giám sát**

b. Data URI Encryption → Bypass kiểm tra tệp tải lên, JSON Wrapping → Bypass kiểm tra dữ liệu không phải JSON, IP Spoofing → Giả mạo địa chỉ IP để tránh giám sát

c. Data URI Encoding → Bypass kiểm tra tệp tải lên, JSON Wrapping → Bypass kiểm tra dữ liệu không phải JSON, IP Fake → Giả mạo địa chỉ IP để tránh giám sát

**Câu 10: Mục đích của kỹ thuật "Null Character Injection" là gì?**

**a. Đưa vào ký tự NULL để bypass kiểm tra của WAF**

b. Chuyển mã hóa dữ liệu thành dạng hex

c. Tạo ra một HTTP request mới

d. Loại bỏ các ký tự đặc biệt khỏi URL

**Câu 11: Ghép các lệnh với tác dụng của chúng trong Command Injection.**

**a. `whoami` → Xác định tên người dùng hiện tại, `ping` → Kiểm tra kết nối mạng tới một địa chỉ IP hoặc tên miền, `ls` → Liệt kê các file và thư mục trong thư mục hiện tại**

b. `pwd` → Xác định tên người dùng hiện tại, `ping` → Kiểm tra kết nối mạng tới một địa chỉ IP hoặc tên miền, `ls` → Liệt kê các file và thư mục trong thư mục hiện tại

c. `whoami` → Xác định tên người dùng hiện tại, `ping` → Kiểm tra kết nối mạng tới một địa chỉ IP hoặc tên miền, `ls /` → Liệt kê các file và thư mục trong thư mục hiện tại

**Câu 12: Ghép các kỹ thuật bảo mật với mô tả của chúng trong ngăn chặn Command Injection.**

**a. Using Prepared Statements → Tránh việc chèn trực tiếp chuỗi vào lệnh, Input Sanitization → Loại bỏ các ký tự không hợp lệ trong đầu vào của người dùng, Escaping Special Characters → Thêm ký tự escape trước các ký tự đặc biệt**

b. Using Prepared Statements → Tránh việc chèn trực tiếp chuỗi vào lệnh, Input Sanitization → Loại bỏ các ký tự không hợp lệ trong đầu vào của người dùng, Encryption Special Characters → Thêm ký tự escape trước các ký tự đặc biệt

c. Using Prepared Statements → Tránh việc chèn trực tiếp chuỗi vào lệnh, Output Sanitization → Loại bỏ các ký tự không hợp lệ trong đầu vào của người dùng, Escaping Special Characters → Thêm ký tự escape trước các ký tự đặc biệt

**Câu 13: Kỹ thuật nào dưới đây có thể được sử dụng để bypass WAF trong tấn công SQL Injection?**

**a. URL encoding**

b. Data compression

c. Tokenization

d. File encryption

**Câu 14: Kỹ thuật "Double URL Encoding" có thể giúp bypass WAF như thế nào?**

**a. Mã hóa URL hai lần để tránh bị phát hiện bởi WAF**

b. Loại bỏ các tham số không cần thiết trong URL

c. Giảm dung lượng yêu cầu HTTP

d. Mã hóa dữ liệu HTTP ở cấp độ ứng dụng

**Câu 15: Mục tiêu của Case Transformation trong SQL Injection là gì?**

**a. Thay đổi chữ hoa và chữ thường để tránh phát hiện**

b. Loại bỏ ký tự NULL khỏi chuỗi

c. Mã hóa toàn bộ URL

d. Che giấu thông tin người dùng

**Câu 16: WAF nào dưới đây có đặc điểm là cookie chứa "TSxxxxxxxx"?**

**a. F5 Big IP**

b. Citrix ADC

c. Modsecurity

d. Imperva Incapsula

**Câu 17: Mô hình bảo mật Positive Security cho phép gì?**

**a. Chỉ cho phép dữ liệu tốt và từ chối tất cả dữ liệu khác**

b. Cho phép tất cả dữ liệu và chỉ từ chối dữ liệu xấu

c. Không kiểm tra dữ liệu đầu vào

d. Tạo ra các yêu cầu HTTP mới

**Câu 18: Kỹ thuật "Base64 Encoding" thường được sử dụng trong bypass WAF để làm gì?**

**a. Mã hóa chuỗi đầu vào để qua mặt kiểm tra của WAF**

b. Tạo một kết nối mới với máy chủ

c. Mã hóa dữ liệu trước khi lưu trữ

d. Chỉ sử dụng cho mục đích phân tích dữ liệu

**Câu 19: Parameter Pollution giúp bypass WAF bằng cách nào?**

**a. Thêm nhiều tham số vào URL để làm khó nhận diện**

b. Mã hóa toàn bộ URL

c. Loại bỏ các ký tự đặc biệt

d. Chia nhỏ HTTP request thành nhiều phần

**Câu 20: Tại sao "Reverse Proxy" thường được kết hợp với WAF?**

**a. Để kiểm soát lưu lượng truy cập và cung cấp thêm lớp bảo mật**

b. Để tăng tốc độ truyền tải dữ liệu

c. Để thay đổi cấu trúc URL của yêu cầu

d. Để mã hóa dữ liệu trước khi lưu trữ

**Câu 21: Ghép các lệnh phổ biến với kết quả đầu ra trong Command Injection.**

**a. `echo Hello` → In ra chuỗi "Hello", `cat /etc/passwd` → Hiển thị nội dung file người dùng trong hệ thống Linux, `pwd` → In ra thư mục hiện tại**

b. `echo Hello` → In ra chuỗi "Hello", `cat /etc/passwd` → Hiển thị nội dung mật khẩu của người dùng trong hệ thống Linux, `pwd` → In ra thư mục hiện tại

c. `echo Hello` → Xuất ra file"Hello", `cat /etc/passwd` → Hiển thị nội dung file người dùng trong hệ thống Linux, `pwd` → In ra thư mục hiện tại

**Câu 22: Kỹ thuật "Case Transformation" trong bypass WAF là gì?**

**a. Biến đổi chữ hoa và chữ thường để tránh nhận diện của WAF**

b. Chuyển đổi các ký tự thành định dạng số

c. Gửi nhiều yêu cầu cùng lúc

d. Che giấu dữ liệu đầu vào bằng mã hóa AES

**Câu 23: Ghép các kỹ thuật bypass với phương pháp tấn công mà chúng hỗ trợ.**

**a.  Null Character Injection → Bypass kiểm tra chuỗi trong Command Injection, Parameter Pollution → Thêm tham số vào URL để bypass kiểm tra WAF, Case Transformation → Thay đổi chữ hoa và chữ thường để bypass SQL Injection**

b.  Null Character Injection → Bypass kiểm tra chuỗi trong Command Injection, Parameter Pollution → Thêm tham số vào URI để bypass kiểm tra WAF, Case Transformation → Thay đổi chữ hoa và chữ thường để bypass SQL Injection

c.  Null Pointer Injection → Bypass kiểm tra chuỗi trong Command Injection, Parameter Pollution → Thêm tham số vào URL để bypass kiểm tra WAF, Case Transformation → Thay đổi chữ hoa và chữ thường để bypass SQL Injection

**Câu 24: Ghép các công cụ với công dụng của chúng trong phân tích Command Injection.**

**a. sqlmap → Tự động kiểm tra và khai thác SQL Injection, Nmap → Quét cổng và phát hiện dịch vụ trên mạng, Burp Suite → Chặn và phân tích lưu lượng HTTP**

b. sqlite3 → Tự động kiểm tra và khai thác SQL Injection, Nmap → Quét cổng và phát hiện dịch vụ trên mạng, Burp Suite → Chặn và phân tích lưu lượng HTTP

c. sqlmap → Tự động kiểm tra và khai thác SQL Injection, Nmab → Quét cổng và phát hiện dịch vụ trên mạng, Burp Suite → Chặn và phân tích lưu lượng HTTP

**Câu 25: Ghép các kỹ thuật Command Injection với mục đích của chúng.**

**a. IP Spoofing → Giả mạo địa chỉ IP trong các yêu cầu mạng, Base64 Encoding → Mã hóa lệnh để tránh phát hiện, File Inclusion → Chèn một file từ hệ thống vào lệnh**

b. IP Spoofing → Giả mạo địa chỉ IP trong các yêu cầu mạng, Base64 Decryption → Mã hóa lệnh để tránh phát hiện, File Inclusion → Chèn một file từ hệ thống vào lệnh

c. IP Spoofing → Giả mạo địa chỉ IP trong các yêu cầu mạng, Base64 Decoding → Mã hóa lệnh để tránh phát hiện, File Inclusion → Chèn một file từ hệ thống vào lệnh

**Câu 26: Một đặc điểm nhận dạng của WAF "Imperva" là gì?**

**a. Cookie chứa chuỗi "incap\_ses\_" hoặc "visid\_incap\_"**

b. Cookie chứa chuỗi "\_cfduid"

c. Header HTTP chứa chuỗi "TSxxxxxxxx"

d. Header HTTP chứa "X-Forwarded-For"

**Câu 27: Fingerprinting WAF có mục đích gì?**

**a. Xác định loại WAF đang bảo vệ ứng dụng web**

b. Thực thi mã độc từ xa

c. Chặn các yêu cầu không hợp lệ

d. Giám sát các thay đổi dữ liệu đầu vào

**Câu 28: WAF nào thường được sử dụng trong môi trường AWS?**

**a. AWS WAF**

b. Cloudflare

c. F5 Big IP

d. Akamai Kona

**Câu 29: Kỹ thuật nào sau đây có thể bypass việc kiểm tra khoảng trắng trong SQLi?**

**a. Inline comment**

b. Data masking

c. Binary encoding

d. Multithreading

**Câu 30: Ghép các kỹ thuật bypass với loại tấn công phù hợp.**

**a. Case Transformation → Bypass kiểm tra SQL Injection, Obfuscation → Che giấu mã độc trong XSS, Inline Comment → Loại bỏ khoảng trắng trong SQL Injection**

b. Case Transformation → Bypass kiểm tra SQL Injection, Obfuscation → Che giấu mã độc trong XLSS, Inline Comment → Loại bỏ khoảng trắng trong SQL Injection

c. Case Transformation → Bypass kiểm tra SQL Injection, Obfuscation → Che giấu mã độc trong XSS, Inline Comment → Loại bỏ chữu thường trong SQL Injection