HTTP Status Code CheatSheet

(in cybersecurity terms)

موضوع :

HTTP Status Code cheat sheet (in cybersecurity terms)

تهیه شده توسط تیم TryHackBox

The Chaos

لینک زیر جهت حمایت از جامعه آموزش رایگان :

https://daramet.com/TryHackBox

ما را به دوستانتان معرفی کنید .

دیگر کانال ها و شبکه های اجتماعی ما را دنبال کنید:

کانال های تلگرام ما

آموزش تست نفوذ و Red Team :

@TryHackBox

رودمپ های مختلف:

@TryHackBoxOfficial

داستان های هک:

@TryHackBoxStory

آموزش برنامه نویسی:

@TryCodeBox

رادیو زیروپاد (پادکست ها):

@RadioZeroPod

اینستاگرام :

http://www.instagram.com/TryHackBox

یوتیوب:

https://youtube.com/@tryhackbox

HTTP Status Code چیست؟

کدهای وضعیت HTTP Status Codes یا HTTP روشی هستند که توسط وب‌سروها به مرورگرها ارسال می‌شوند تا نتیجه‌ی درخواست‌های ارسالی از سمت کاربران (مثل درخواست برای باز کردن یک صفحه وب) را مشخص کنند. هر کد وضعیت، پیامی خاص دارد و به کاربر یا توسعه‌دهنده اطلاع می‌دهد که درخواست آن‌ها چه نتیجه‌ای داشته است.

کدهای وضعیت HTTP در 5 دسته‌بندی کلی قرار می‌گیرند:

- کدهای 1xx (اطلاعاتی): این کدها نشان‌دهنده دریافت درخواست توسط سرور و ادامه فرآیند هستند.
- کدهای 2xx (موفقیت آمیز): این کدها نشان‌دهنده موفقیت‌آمیز بودن درخواست هستند.
- کدهای 3xx (تغییر مسیر): این کدها به کاربر اطلاع می‌دهند که باید اقدامات بیشتری معمولاً انتقال به آدرس URL دیگری انجام دهد.
- کدهای 4xx (خطای کاربر): این کدها خطاهایی هستند که به دلیل مشکلی در سمت‌کاربر حادث شده‌اند.
- کدهای 5xx (خطای سرور): این کدها خطاهایی هستند که به دلیل مشکلی در سمت سرور حادث شده‌اند.

کدهای وضعیت 1xx (Informational)

- **100 Continue**

درخواست اولیه را دریافت کرده و کاربر می‌تواند ادامه داده‌ها را ارسال کند.

نقش در تست نفوذ:

- اگر سرور به طور مداوم این کد را ارسال کند، ممکن است نشانه‌ای از پیکربندی اشتباه یا رفتار غیرعادی سرور باشد.
- میتواند در حملات Slowloris (حملات انکار سرویس با اتصالات طولانی) مورد بررسی قرار گیرد.

- **101 Switching Protocols**

اطلاع می‌دهد که درخواست تغییر پروتکل را قبول کرده و تغییر به پروتکلی که در هدر Upgradeمشخص شده است، انجام خواهد شد.

نقش در تست نفوذ:

- تغییر پروتکل (مثلاً از HTTP به WebSocket) ممکن است حملات Man-in-the-Middle (MITM) را تسهیل کند اگر پروتکل جدید امنیت کافی نداشته باشد.

هکرها ممکن است از این کد برای آزمایش آسیب‌پذیری‌های ارتقای پروتکل استفاده کنند.

مثال: کد 101 Switching Protocols ممکن است نشان دهد سرور از پروتکل‌های قدیمی یا غیرایمن مثل WebSocket بدون رمزگذاری پشتیبانی می‌کند.

- **102 Processing (WebDAV)**

این کد نشان می‌دهد که سرور درخواست را دریافت کرده و در حال پردازش آن است، اما هنوز هیچ پاسخی آماده نشده است.

نقش در تست نفوذ:

- نشان‌دهنده استفاده از WebDAV است که اگر به درستی پیکربندی نشود، میتواند منجر به دسترسی غیرمجاز به فایل‌ها شود.
- هکرها ممکن است این کد را به عنوان نشانه‌ای از سرویس‌های حساس (مثل سیستم‌های مدیریت محتوا) بررسی کنند.

- **103 Early Hints**

این کد توسط سرور ارسال می‌شود تا قبل از آماده‌سازی پاسخ نهایی، اطلاعاتی درباره منابع و سیاست‌هایی که انتظار می‌رود در پاسخ نهایی به آن‌ها ارجاع داده شود، ارائه دهد. این امکان را به مرورگر می‌دهد تا حتی قبل از دریافت پاسخ نهایی، با سیاست‌ها ارتباط برقرار کند یا منابع را پیش‌بارگیری (preload) کند.

نقش در تست نفوذ:

- اطلاعات پیش‌بارگیری (preload) ممکن است مسیرهای مخفی یا منابع حساس را فاش کند.
- میتواند در شناسایی ساختار داخلی سرور یا سیاست‌های امنیتی آن مفید باشد.

نکته:

این کدهای وضعیت بیشتر در موارد خاص و برای ارتباطات مبتنی بر پروتکل‌های ویژه کاربرد دارند. معمولاً برای توسعه‌دهندگان وب و کسانی که به جزئیات فنی عمیق‌تری نیاز دارند، اهمیت دارند.

کد های وضعیت 2xx (Success)

- **200 OK**

این کد رایج‌ترین پاسخ موفقیت‌آمیز است و نشان می‌دهد که درخواست به درستی دریافت، فهمیده شده و پردازش شده است.

نقش در تست نفوذ:

- تایید موفقیت‌آمیز بودن حمله مثلاً SQL Injection یاXSS
- شناساییendpoint های فعال و حساس
- تشخیص پیکربندی‌های نادرست (مثلاً دسترسی به صفحات مدیریت بدون احراز هویت)

- **201 Created**

درخواست موفقیت‌آمیز بوده و منجر به ایجاد منبع جدید شده است. این کد معمولاً پس از ارسال درخواست‌های POST که منجر به ایجاد منابع جدید می‌شوند، استفاده می‌گردد.

نقش در تست نفوذ:

- شناسایی قابلیت ایجاد منابع جدید (مثل کاربران یا فایل‌ها)
- امکان سوءاستفاده ازAPI های حساس
- تشخیص مسیرهای upload غیرایمن

- **202 Accepted**

درخواست پذیرفته شده است، اما پردازش آن هنوز تکمیل نشده است.

این کد در مواردی که پردازش درخواست زمان‌بر است (مثل عملیات پس‌زمینه) استفاده می‌شود.

نقش در تست نفوذ:

- شناسایی سیستم‌های پردازش ناهمزمان
- امکان حملات Race Condition
- تشخیص عملیات‌های زمان‌بر که ممکن است آسیب‌پذیر باشند

- 203 Non-Authoritative Information

پاسخ ارسالی حاوی اطلاعاتی است که از منبعی غیر از سرور اصلی جمع‌آوری شده است (مثل یک پروکسی یا کش).

- **204 No Content**

سرور درخواست را پردازش کرده است، اما محتوایی برای ارسال به کاربر ندارد.

نقش در تست نفوذ:

- تایید عملیات مخرب بدون بازخورد مثل DELETE موفق
- شناسایی endpoint های حساس که پاسخ نمی‌دهند
- امکان حملات Blind مثل Blind SQL Injection

- **205 Reset Content**

سرور به کاربر دستور می‌دهد که محتوای نمایش‌داده‌شده را ریست کند (مثل پاک کردن فرم پس از ارسال).

- **206 Partial Content**

این کد برای درخواست‌هایی استفاده می‌شود که فقط بخشی از یک منبع را درخواست کرده‌اند (مثل دانلودهای تکه‌تکه یا ادامه‌دار).

نقش در تست نفوذ:

- امکان دانلود تکه‌تکه فایل‌های حساس
- دور زدن محدودیت‌های دسترسی
- شناسایی قابلیت Range Request برای حملات خاص

- **207 Multi-Status (WebDAV)**

پاسخ شامل مجموعه‌ای از وضعیت‌های مختلف است که به صورت XML ارائه می‌شود.

در پروتکل WebDAV استفاده می‌شود تا وضعیت چندین عملیات را در یک پاسخ گزارش دهد.

نقش در تست نفوذ:

- شناسایی سرویس‌های WebDAV فعال
- امکان سوءاستفاده از قابلیت‌های پیشرفته
- تشخیص ساختار فایل‌سیستم سرور

- **208 Already Reported (WebDAV)**

نشان می‌دهد بخشی از منبع درخواست‌شده قبلاً در پاسخ دیگری گزارش شده است.

برای جلوگیری از تکرار اطلاعات در پاسخ‌های متعدد استفاده می‌شود.

نکات کلیدی:

کدهای 202 و 204 برای عملیات ناهمگام (Async) مفیدند.

کدهای 206 و 207 در سیستم‌های پیشرفته مانند WebDAV یا دانلود مدیران فایل کاربرد دارند.

کد 206 ممکن است نشان‌دهنده استفاده از کش یا پروکسی باشد.

استراتژی‌های هکرها برای استفاده از کد های موفقیت آمیز

1. **شناسایی ساختار سیستم:**
   - استفاده از پاسخ‌های 2xx برای نقشه‌برداری از سیستم
   - تشخیص API ها و endpoint های فعال
2. **تایید حملات موفق:**
   - پاسخ 200 پس از ارسال payload مخرب
   - پاسخ 201 پس از ایجاد منابع غیرمجاز
3. **سوءاستفاده از قابلیت‌ها:**
   - استفاده از 206 برای استخراج اطلاعات حساس
   - سوءاستفاده از 207 برای شناسایی ساختار فایل‌سیستم

کد های وضعیت 3xx (Redirection)

- **300 Multiple Choices**

چندین گزینه برای منبع مورد نظر وجود دارد و کاربر/کلاینت باید یکی را انتخاب کند.

مثال: نمایش نسخه‌های مختلف یک فایل (مثل فرمت‌های PDF یا DOCX).

- **301 Moved Permanently**

منبع به طور دائم به آدرس جدید منتقل شده است.

رفتار مرورگر: مرورگرها به صورت خودکار به URL جدید هدایت می‌شوند و این تغییر را ذخیره می‌کنند.

کاربرد: تغییر ساختار وبسایت بدون از دست دادن ترافیک.

- **302 Found**

منبع موقتاً به آدرس دیگری منتقل شده است.

تفاوت با 301: مرورگرها این تغییر را ذخیره نمی‌کنند و در آینده دوباره به URL اصلی مراجعه می‌کنند.

مثال: هدایت کاربر پس از لاگین موقت به صفحه اصلی.

- **303 See Other**

کاربر باید برای دریافت پاسخ به URL دیگری مراجعه کند.

کاربرد اصلی: معمولاً پس از ارسال فرم (POST) استفاده می‌شود تا کاربر به صفحه نتیجه هدایت شود.

ویژگی کلیدی: متد درخواست به GET تغییر می‌کند (برخلاف 308/307).

- **304 Not Modified**

نسخه کش‌شده منبع با نسخه سرور یکسان است و نیاز به دانلود مجدد نیست.

مکانیزم: با هدرهای If-Modified-Since یا ETag کار می‌کند.

فواید: کاهش ترافیک شبکه و بهبود عملکرد.

- **307 Temporary Redirect**

تغییر مسیر **موقت** با حفظ متد اصلی درخواست.(POST/GET)

تفاوت با 302: تضمین می‌کند متد درخواست تغییر نمی‌کند امن‌تر برای POST

- 308 Permanent Redirect

تغییر مسیر **دائم** با حفظ متد اصلی درخواست.

تفاوت با 301: مانند 307، متد درخواست را حفظ می‌کند.

مثال: انتقال دائم یک فرم POST به آدرس جدید.

استراتژی هکر ها برای استفاده از کد های تغییر مسیر

کدهای تغییر مسیر می‌توانند برای هکرها نقاط حمله جذابی ایجاد کنند:

301 Moved Permanently / 308 Permanent Redirect

- **تهدیدات:**
  - سوءاستفاده از تغییر مسیرهای دائمی برای:
    - پنهان کردن فعالیت‌های مخرب (مثل ریدایرکت به سایت فیشینگ)
    - دور زدن سیستم‌های امنیتی (با ریدایرکت از دامنه‌های معتبر به دامنه‌های مخرب)
  - امکان cache poisoning در مرورگر قربانی

302 Found / 307 Temporary Redirect

- **تهدیدات:**
  - حملات Open Redirect ابزاری کلیدی برای فیشینگ:

http://legitsite.com/redirect?url=evil.com

  - سوءاستفاده از ریدایرکت‌های موقت برای:
    - دور زدن WAF ها و سیستم‌های تشخیص نفوذ
    - پنهان کردن ترافیک مخرب

303 See Other

- **تهدیدات:**
  - تغییر متد POST به GET که ممکن است:
    - اطلاعات حساس را در URL نمایان کند
    - امکان سوءاستفاده از حملات CSRF را افزایش دهد

304 Not Modified

- **تهدیدات:**
  - شناسایی منابع قابل کش شدن) برای حملات(cache poisoning
  - تشخیص سیستم‌هایی که از اعتبارسنجی ETag ضعیفی استفاده می‌کنند

300 Multiple Choices

- **تهدیدات:**
  - شناسایی endpoint های حساس که چندین فرمت خروجی دارند
  - امکان کشف فایل‌های پشتیبان) مثل(index.php.bak

تکنیک‌های متداول حمله:

1. **حملات Open Redirect**
   - استفاده از پارامترهای کنترل‌نشده در URL برای ریدایرکت به سایت‌های مخرب
   - مثال:

https://victim.com/logout?redirect=https://evil.com

2. **حملات Cache Poisoning**
   - تزریق پاسخ‌های مخرب به کش سرور یاCDN
   - استفاده از 304/301 برای ماندگاری بیشتر پاسخ‌های آلوده
3. **دور زدن احراز هویت:**
   - سوءاستفاده از ریدایرکت‌ها برای دسترسی بهendpoint های داخلی

اقدامات دفاعی:

1. **اعتبارسنجی دقیقURL های ریدایرکت:**
   - فقط اجازه ریدایرکت به دامنه‌های معتبر
   - استفاده از لیست سفید (whitelist) برای آدرس‌های مجاز
2. **محدودیت‌های امنیتی:**
   - غیرفعال کردن ریدایرکت برایendpoint های حساس
   - پیاده‌سازی هدرهای امنیتی مثل:

```
Content-Security-Policy: default-src 'self'
```

3. **مانیتورینگ:**
   - ثبت تمام درخواست‌های ریدایرکت در لاگ‌ها
   - هشدار برای ریدایرکت‌های غیرمعمول
4. **مقابله با کش‌پوینسونینگ:**
   - استفاده از Vary: Origin و Vary: Referer
   - محدود کردن مدت زمان کش پاسخ‌ها

کد های وضعیت 4xx (Client Eror)

- **400 Bad Request**

سرور نتوانست درخواست را به دلیل ساختار نادرست پردازش کند.

علل شایع :پارامترهای نامعتبر، JSONناقص، هدرهای نادرست.

- **401 Unauthorized**

برای دسترسی به منبع، احراز هویت لازم است.

تفاوت با 403 :کاربر هنوز هویت خود را اثبات نکرده است.

- **403 Forbidden**

سرور درخواست را فهمیده، اما دسترسی را ممنوع کرده است.

مثال :کاربری که به صفحه مدیریت دسترسی ندارد.

- **404 Not Found**

منبع درخواستی روی سرور وجود ندارد.

تهدید امنیتی :ممکن است نشانگر مسیرهای حذف‌شده یا مخفی باشد.

- **405 Method Not Allowed**

متد استفاده‌شده مثل PUT یا DELETE برای این منبع مجاز نیست.

کاربرد :شناسایی متدهای فعال درAPI ها.

- **406 Not Acceptable**

سرور نمی‌تواند پاسخ مطابق با هدر Acceptکاربر ارائه دهد.

مثال :درخواست فرمت XML در حالی که سرور فقط JSON پشتیبانی می‌کند.

- **408 Request Timeout**

سرور در زمان تعیین‌شده پاسخ را دریافت نکرد.

تهدید :ممکن است در حملات Slowloris استفاده شود.

- **409 Conflict**

درخواست با وضعیت فعلی سرور تداخل دارد.

مثال :ویرایش همزمان یک سند توسط دو کاربر.

- **410 Gone**

منبع به‌صورت دائم حذف شده است (متفاوت با 404).

سرور تأیید می‌کند که منبع قبلاً وجود داشته است.

- **423 Locked (WebDAV)**

منبع قفل شده و قابل ویرایش نیست.

کاربرد :در سیستم‌های مدیریت محتوای مشترک.

- **429 Too Many Requests**

کاربر در بازه زمانی مشخص تعداد زیادی درخواست ارسال کرده است.

دفاع :مکانیزمی برای جلوگیری از حملات Brute-Force.

استراتژی هکر ها برای استفاده از کد های خطای سمت کاربر

برای هکرها حکم **نقشه گنج** را دارند و اطلاعات حیاتی درباره سیستم هدف فاش می‌کنند:

400 Bad Request

- **نقش در نفوذ:**
  - شناسایی **پارامترهای حساس** با ارسال داده‌های نادرست
  - تشخیص **آسیب‌پذیری‌های ورودی** مثل SQL Injection یا XSS
  - مثال:

```
POST /login HTTP/1.1 {"username":"admin'--","password":""}
```

401 Unauthorized

- **نقش در نفوذ:**
  - شناسایی **صفحات احراز هویت**
  - تشخیص مکانیزم‌های امنیتی مثل Basic Auth یاJWT
  - هدف قرار دادنendpoint های حساس:

```
GET /admin HTTP/1.1
```

403 Forbidden

- **نقش در نفوذ:**
  - کشف **مسیرهای مخفی** مثل /admin ، /backup
  - تشخیص **دسترسی‌های نادرست** مثلDirectory Listing
  - حملات **Bypass**با تکنیک‌هایی مثل:

```
GET /admin/../admin HTTP/1.1
```

404 Not Found

- **نقش در نفوذ:**
  - **مسیریابی حمله** با تفاوت پاسخ‌ها:
    - پاسخ 404 برای /secret/ vs پاسخ 403 برای /secret/
  - شناسایی **فایل‌های پشتیبان:**

```
GET /index.php.bak HTTP/1.1
```

405 Method Not Allowed

- **نقش در نفوذ:**
  - شناسایی **متدهای فعال** مثل PUT/DELETE
  - تشخیص **API های ناامن:**

```
OPTIONS /api/users HTTP/1.1
```

429 Too Many Requests

- **نقش در نفوذ:**
  - تشخیص **محدودیت نرخ درخواست** برای حملات Brute-Force
  - شناسایی **آستانه‌های امنیتی** سیستم

تکنیک‌های پیشرفته هکرها:

1. **تحلیل تفاوت پاسخ‌ها:**
   - مقایسه پاسخ‌های 401 vs 403 برای تشخیص ساختار احراز هویت
   - تفاوت 404 vs 403 برای یافتن مسیرهای حساس

2. **سوءاستفاده از خطاها:**
   - استفاده از 400 برای **فاز تشخیص** حملات تزریقی
   - سوءاستفاده از 405 برای **کشف متدهای خطرناک**
3. **حملات انکار سرویس:**
   - تحریک پاسخ‌های 429 برای **شناسایی محدودیت‌ها**
   - استفاده از 408 برای **اتصال‌های طولانی‌مدت**

اقدامات دفاعی برای مدیران سیستم:

1. **پیکربندی امن:**
   - غیرفعال کردن متدهای غیرضروری(PUT/DELETE)
   - محدود کردن اطلاعات خطا) غیرفعال کردن(stack trace
2. **مانیتورینگ:**
   - ثبت تمام درخواست‌های منجر به xx4
   - هشدار برای الگوهای غیرعادی (مثل 403 مکرر)
3. **مقابله فنی:**
   - پیاده‌سازی **WAF**برای فیلتر درخواست‌های مخرب
   - استفاده از **rate limiting**هوشمند

مثال واقعی:

هکری با تحلیل پاسخ‌های 403 متوجه می‌شود مسیر /wp-admin وجود دارد، سپس با ارسال هدرهای جعلی (X-Forwarded-For: 127.0.0.1) دسترسی مدیریتی به دست می‌آورد.

این کدها اگرچه نشانگر خطای کاربر هستند، اما **شناسنامه سیستم** محسوب می‌شوند!

کدهای وضعیت 5xx (Server Error)

این کدها به توسعه‌دهندگان اطلاع می‌دهند که مشکل در سمت سرور وجود دارد و نیاز به بررسی و رفع دارد. درک این کدها به تیم‌های فنی کمک می‌کند تا سریعتر مشکلات سرور را تشخیص داده و سیستم‌های خود را پایدار و در دسترس نگه دارند.

کدهای اصلی خطای سرور:

- **500 Internal Server Error**

نشان می‌دهد سرور با یک خطای داخلی مواجه شده و نمی‌تواند درخواست را پردازش کند.

مثال :خطا در اسکریپت‌های backend یا پیکربندی نادرست سرور.

نقش در تست نفوذ:

- نشان‌دهنده **خطای عمومی** در سرور است که ممکن است ناشی از **پیکربندی اشتباه**، **اسکریپت‌های معیوب** یا **آسیب‌پذیری‌های نرم‌افزاری** باشد.
- هکرها با ارسال درخواست‌های مخرب مثل SQL Injection یا Path Traversal این کد را تحریک می‌کنند تا **حفره‌های امنیتی** را شناسایی کنند.
- اگر خطای 500 همراه با **جزئیات خطا** مثل stack trace نمایش داده شود، ممکن است **اطلاعات حساس** (مثل ساختار دیتابیس یا کد منبع) را فاش کند.

- **501 Not Implemented**

سرور قادر به انجام درخواست نیست زیرا عملکرد یا ویژگی مورد نیاز پشتیبانی نمی‌شود.

مثال :درخواست استفاده از متد HTTP ناشناخته برای سرور.

نقش در تست نفوذ:

- نشان می‌دهد سرور از متد درخواستی پشتیبانی نمی‌کند (مثل درخواست PUT روی سروری که فقط GET را قبول می‌کند).

هکرها از این کد برای شناسایی متدهای غیرفعال استفاده می‌کنند که ممکن است در صورت فعال شدن، خطرناک باشند (مثل متدهای DELETE یا TRACE).

- **502 Bad Gateway**

هنگامی رخ می‌دهد که سرور به عنوان گیتوی یا پروکسی، پاسخ نامعتبری از سرور بالادست دریافت کند.

مثال :قطعی ارتباط بین سرور اصلی و سرور میانی.

نقش در تست نفوذ:

- معمولاً در معماری‌های Load Balancer یا Reverse Proxy دیده می‌شود.
- ممکن است نشان‌دهنده آسیب‌پذیری در زیرساخت باشد (مثل سرورهای backend در معرض حمله).

هکرها از این کد برای شناسایی مسیرهای غیرمستقیم به سرورهای داخلی استفاده می‌کنند.

- **503 Service Unavailable**

سرور موقتاً در دسترس نیست، معمولاً به دلیل overload یا تعمیرات.

مثال :ترافیک بسیار بالا در زمان فروش ویژه.

نقش در تست نفوذ:

- نشان‌دهنده ترافیک بالا یا تحت حمله بودن سرور است (مثل حملات DDoS).

هکرها ممکن است از این کد برای تشخیص حملات موفق یا ضعف در مقابله با overload استفاده کنند.

اگر سرور پس از 503، اطلاعات Retry-After را نمایش دهد، ممکن است زمانبندی حملات را تنظیم کنند.

- **504 Gateway Timeout**
  - سرور به عنوان گیتوی یا پروکسی، در زمان مشخص پاسخی از سرور بالادست دریافت نکرده است.
  - **مثال :**زمان‌بندی نامناسب برای پاسخ‌دهی بین سرورها.

نقش در تست نفوذ:

نشان می‌دهد سرور پاسخی از سرور بالادستی دریافت نکرده است.

می‌تواند نشانه‌ای از آسیب‌پذیری در ارتباط بین سرورها باشد (مثل Slowloris Attack).

- **505 HTTP Version Not Supported**
  - نسخه HTTP استفاده شده در درخواست، توسط سرور پشتیبانی نمی‌شود.
  - **مثال :**درخواست با HTTP/3 روی سروری که فقط HTTP/1.1 را ساپورت می‌کند.

نقش در تست نفوذ:

- اگر سرور نسخه‌های قدیمی HTTP (مثل HTTP/1.0) را رد کند، هکرها ممکن است از آن برای شناسایی پروتکل‌های ناامن استفاده کنند.

- **511 Network Authentication Required**

کاربر باید برای دسترسی به شبکه احراز هویت شود (مثلاً در وای‌فای عمومی).

مثال :صفحه لاگین هتل قبل از دسترسی به اینترنت.

استراتژی هکر ها برای استفاده از کد های خطای سرور

1. **شناسایی آسیب‌پذیری‌ها:**

- ارسال درخواست‌های مخرب مثل /../etc/passwd و بررسی پاسخ 500 برای یافتن **Path Traversal**
- تست ورودی‌های کاربر مثل <script> برای یافتن **XSS** یا **SQL Injection**

2. **نقاط ضعف زیرساخت:**

- بررسی کدهای 502/564 برای یافتن **سرورهای داخلی آسیب‌پذیر.**
- اسکن پورت‌های باز با استفاده از خطاهای 503

3. **حملات DoS/DDoS:**

- اگر سرور به راحتی به کد 503 می‌رسد، ممکن است هدف خوبی برای **حملات انکار سرویس** باشد.

اقدامات دفاعی برای مدیران سرور:

- **پنهان کردن جزئیات خطاها** غیرفعال کردن نمایش stack trace
- **مانیتورینگ مداوم** پاسخ‌های 5xx برای شناسایی حملات.
- **آپدیت نرم‌افزارها** برای رفع آسیب‌پذیری‌های شناخته شده.
- **محدود کردن متدهای HTTP** به موارد ضروری مثل غیرفعال کردن TRACE

نکات کلیدی:

- کدهای **5xx** عموماً نشان‌دهنده مشکلات **سمت سرور** هستند و نیاز به بررسی فوری دارند.
- کد **503** ممکن است به دلیل حملات DDoS رخ دهد.
- کد **502/504** اغلب در معماری‌های میکروسرویس یا استفاده از CDN دیده می‌شود.
- **توصیه :** لاگ‌گیری منظم از این خطاها به شناسایی الگوهای حملات یا نقاط ضعف سیستم کمک می‌کند.