Mass Exploitation of MOVEit Transfer Critical Vulnerability

Analysis of an incident exploiting CVE-2023-34362

Author: Prashant Tilekar
Sai Molige
Daniel dos Santos

Date: June 8, 2023
## Contents

1. Executive summary .................................................................................................................. 3  
2. Incident analysis ...................................................................................................................... 3  
3. The human2.aspx webshell .................................................................................................... 6  
4. Recommended mitigations ....................................................................................................... 8  
5. IOCs ........................................................................................................................................ 9
1. Executive summary

On May 31, Forescout Research - Vedere Labs uncovered a significant incident where threat actors exploited a critical zero-day vulnerability in the MOVEit Transfer software, which resulted in unauthorized access to and exfiltration of private data, as well as privilege escalation.

MOVEit Transfer is a widely adopted managed file transfer (MFT) solution that enables organizations to securely exchange files with their business partners and customers. The exploited vulnerability has been assigned the identifier CVE-2023-34362.

CVE-2023-34362 is currently being mass exploited, with hundreds of organizations hit simultaneously. Although we could not attribute this particular incident to a specific threat actor with certainty, ongoing exploitation of CVE-2023-34362 has been attributed by CISA, the FBI and other organizations to the Cl0p ransomware group since May 27. The criminal group itself has claimed responsibility for the attacks with an extortion note on their website.

Cl0p is one of the most active ransomware groups and was behind last year’s attack on a UK water utility, among many other critical incidents. The group also exploited another vulnerability in a similar MFT tool in January, claiming 130 victims at that time. Researchers found evidence that the group knew about the MOVEit Transfer vulnerability for almost two years but chose to wait for the right moment to use it in a mass attack.

CVE-2023-34362 is an SQL injection affecting MOVEit Transfer versions prior to 2021.0.6 (13.0.6), 2021.1.4 (13.1.4), 2022.0.4 (14.0.4), 2022.1.5 (14.1.5) and 2023.0.1 (15.0.1). The vulnerability allows attackers to manipulate the underlying database and potentially gain unauthorized access. Exploitation of unpatched systems can occur over both HTTP and HTTPS, making all vulnerable instances susceptible to attack.

Fortunately, the software vendor, Progress, promptly addressed this vulnerability and released a patch to mitigate the risk. On June 9, the vendor also released a patch for a second SQL injection vulnerability (CVE is pending) to address concerns of exploit staging. There is no evidence that the second vulnerability has been exploited in the wild.

There are currently more than 2,500 exposed servers running MOVEit Transfer. Seventy-three percent of those are in the U.S., 5% in the UK and 4.5% in Germany, with the remaining 17.5% spread across over 80 other countries. Sixty-eight percent of the servers have a similar configuration, running over HTTPS on port 443 on top of the Microsoft IIS web server. These servers are most often observed in organizations in the healthcare, financial services and government sectors.

In this report, we present details of the incident we observed, an analysis of a webshell deployed as payload in the incident and recommended mitigations.

2. Incident analysis

The figure below summarizes the incident that we have detected and analyze further below. First, the threat actor exploited CVE-2023-34362 on an Internet-facing host running a vulnerable version of MOVEit Transfer. Second, the attacker deployed a webshell named human2.aspx that allowed them to execute commands on the target. Third, the attacker leveraged the webshell to exfiltrate data to a C2 server.
The table below shows the IIS logs pertaining to the compromised host, which allowed us to further understand the incident.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Details (format: date time source_ip cs-method cs-uri-stem cs-uri-query source_port cs-username c-ip cs(User-Agent) sc-status sc-substatus sc-win32-status time-taken)</th>
</tr>
</thead>
<tbody>
<tr>
<td>POST /machine2.aspx</td>
<td>2023-05-30 18:44:53 ::1 POST /machine2.aspx - 80 - ::1 CWinInetHTTPClient - 200 0 0 59</td>
</tr>
<tr>
<td>GET /human2.aspx</td>
<td>2023-05-30 18:45:12 10.x.x.x GET /human2.aspx - 443 - 5.252.189.191 user-agent - 200 0 0 68</td>
</tr>
<tr>
<td>POST /guestaccess.aspx</td>
<td>2023-05-30 18:44:55 10.x.x.x POST /guestaccess.aspx - 443 - 5.252.190.129 user-agent - 200 0 0 506</td>
</tr>
<tr>
<td></td>
<td>2023-05-30 18:45:00 10.x.x.x POST /guestaccess.aspx - 443 - 5.252.190.129 user-agent - 200 0 0 5216</td>
</tr>
<tr>
<td>POST /api/v1/token</td>
<td>2023-05-31 14:40:36 10.x.x.x POST /api/v1/token - 443 - 155.3.252.241 user-agent 200 0 281</td>
</tr>
<tr>
<td>GET /api/v1/folders</td>
<td>2023-05-30 18:45:02 10.x.x.x GET /api/v1/folders - 443 - 5.252.190.56 user-agent - 200 0 0 184</td>
</tr>
<tr>
<td>POST /api/v1/folders/</td>
<td>2023-05-30 18:45:03 10.x.x.x POST /api/v1/folders/899187381/files uploadType=resumable 443 - 5.252.190.56 user-agent - 200 0 0 175</td>
</tr>
<tr>
<td>POST /moveitisapi/moveitisapi.dll action=m2</td>
<td>2023-05-30 18:44:53 10.x.x.x POST /moveitisapi/moveitisapi.dll action=m2 443 - 5.252.190.186 user-agent - 200 0 0 182</td>
</tr>
<tr>
<td></td>
<td>2023-05-30 18:45:00 10.x.x.x POST /moveitisapi/moveitisapi.dll action=m2 443 - 5.252.190.129 user-agent - 200 0 0 55</td>
</tr>
</tbody>
</table>
**Activity** | **Details**
---|---
**format:** *date time source_ip cs-method cs-uri-stem cs-uri-query source_port cs-username c-ip cs(User-Agent) sc-status sc-substatus sc-win32-status time-taken***

<table>
<thead>
<tr>
<th>Activity</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>2023-05-30 18:45:03 10.x.x.x POST /moveitisapi/moveitisapi.dll action=m2 443 - 5.252.190.56 user-agent - 200 0 0 52</td>
<td></td>
</tr>
<tr>
<td>2023-05-30 18:45:06 10.x.x.x POST /moveitisapi/moveitisapi.dll action=m2 443 - 5.252.190.32 user-agent - 200 0 0 52</td>
<td></td>
</tr>
<tr>
<td>PUT /api/v1/folders/ and files uploadType=resumable&amp;fileId=</td>
<td>2023-05-30 18:45:06 10.x.x.x PUT /api/v1/folders/899187381/files uploadType=resumable&amp;fileId=963078804 443 - 5.252.190.32 user-agent-500 0 0 1352</td>
</tr>
<tr>
<td>GET /moveitisapi/moveitisapi.dll action=capa</td>
<td>2023-05-30 18:44:52 10.x.x.x GET /moveitisapi/moveitisapi.dll action=capa 443 - 5.252.190.34 user-agent-200 0 0 46</td>
</tr>
</tbody>
</table>

**/moveitisapi/moveitisapi.dll** – This is a legitimate DLL with always-present action parameters. The action parameter can have a variety of values, some of which are obvious (such as upload_check, ping, download and end_downld) and others that are less obvious (such as upload_nowiz, capa, hu_downld and m2).

**/guestaccess.aspx** – This is a legitimate DLL that has very limited logs. The only consistent pattern for this, at least in the cluster we observed, is when a request comes from an external IP address, it has 302 response code.

**/api/v1/token** – This is a legitimate endpoint that is responsible for “Get/renew a session token using MOVEit Transfer user credentials.”

**/api/v1/folders** – This is a legitimate endpoint that is responsible for “Folders, folder actions, folder contents and folder properties.” This will have the homeFolderID.

The last two logs in the sample below show the threat actors testing whether the webshell is functioning as expected. One request from 5.252.191.91 returns a 404 error (likely due to the omission of required headers and pass string from the webshell), while the request from 5.252.189.191 returns a 200 response. The order of the
last four logs is also interesting, as /moveitisapi/moveitisapi.dll is used to perform SQL injection while guestacccess.aspx is used for further actions.

/ downloads - We have also observed file and folder downloads from /downloads URI right after the successful testing of the functionality and parameter check of the webshell. The Refer could be either an IP address of the exposed or could be the domain name of MOVEit Transfer. This indicates more of the "smash and grab" to recoup as many files as possible.

3. The human2.aspx webshell

The figures below are all snippets of code from the human2.aspx webshell. At the beginning, it imports legitimate MOVEit DMZ classes and establishes a database connection (in this case, using MySQL) with the provided database settings. It then returns an object indicating the success or failure of the connection. The code uses a class-level Random object called random, along with a RandomString method that generates a random string of a specified length, used later in username creation.

Once connected, the Page_load function checks the value of the received X-siLock-Comment header against a pre-defined password. If the provided password fails to match the expected value, a 404 response code is immediately returned. If the password is correct, then the webshell establishes a successful connection with the server and provides functionality according to the value of the X-siLock-Step1 header, which can be -1, -2 or null.
If the value of `X-siLock-Step1` is -1, the webshell leaks Azure information and gathers crucial data from the MOVEit environment. This involves leveraging the response header to expose Azure-related details and generating a GZIP stream that includes files, their owners, sizes and institutional data stored within MOVEit.

If the value of `X-siLock-Step1` is -2, the webshell deletes a user named "Health Check Service" from the database.

If no specific `X-siLock-Step1` value is specified, the webshell retrieves files specified by the `X-siLocked-Step2` and `X-SiLocked-Step3` headers, enabling the seamless transfer of specific files, as requested.

If the `X-siLocked-Step2` and `X-SiLocked-Step3` headers are not provided, then the webshell introduces a new administrative user named "Health Check Service" into the database. This new user is assigned administrative privileges, granting the attacker elevated access within the system.
4. **Recommended mitigations**

Progress, the MOVEit Transfer vendor, has released immediate mitigation measures to assist in preventing the exploitation of CVE-2023-34362. The table below shows the security patch for each supported version of MOVEit Transfer. Customers on unsupported versions should upgrade to one of the supported fixed versions below.

<table>
<thead>
<tr>
<th>Affected Version</th>
<th>Fixed Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOVEit Transfer 2023.0.0 (15.0)</td>
<td>MOVEit Transfer 2023.0.1</td>
</tr>
<tr>
<td>MOVEit Transfer 2022.1.x (14.1)</td>
<td>MOVEit Transfer 2022.1.5</td>
</tr>
<tr>
<td>MOVEit Transfer 2022.0.x (14.0)</td>
<td>MOVEit Transfer 2022.0.4</td>
</tr>
<tr>
<td>MOVEit Transfer 2021.1.x (13.1)</td>
<td>MOVEit Transfer 2021.1.4</td>
</tr>
<tr>
<td>MOVEit Transfer 2021.0.x (13.0)</td>
<td>MOVEit Transfer 2021.0.6</td>
</tr>
<tr>
<td>MOVEit Transfer 2020.1.x (12.1)</td>
<td>Special patch available</td>
</tr>
<tr>
<td>MOVEit Transfer 2020.0.x (12.0) or older</td>
<td>Must upgrade to a supported version</td>
</tr>
<tr>
<td>MOVEit Cloud</td>
<td>Prod: 14.1.4.94 or 14.0.3.42 Test: 15.0.1.37</td>
</tr>
</tbody>
</table>

Additional recommended mitigation includes:

- Disable all HTTP and HTTPS traffic to your MOVEit Transfer environment. For instance, modify firewall rules to deny HTTP and HTTPS traffic towards affected products on ports 80 and 443.
- Review logs for unexpected downloads of files from unknown IPs or large numbers of files downloaded. Give special attention to GET requests with the cs_uri_stem=/download parameter. These requests may
indicate attempts at file exfiltration, where unauthorized individuals or threat actors are attempting to retrieve sensitive data from the system.

- Delete unauthorized files (such as human2.aspx) and user accounts (such as “Health Check Service”) found on a system.

5. IOCs

The following IOCs have been observed either as part of the incident we analyzed or from external public sources. Most of the observed IP addresses have SSH, OpenVPN, GhostVPN or HTTP proxies open, with some exceptions that have RDP, SMB, NetBios and routers. The latter services are likely compromised devices on their own, but we did not find evidence to back up the claim.

<table>
<thead>
<tr>
<th>Type</th>
<th>Indicators</th>
</tr>
</thead>
</table>
| IP address | 92.118.36.112  
             | 92.51.2.10  
             | 3.132.217.53  
             | 186.211.1.7  
             | 5.252.189.191  
             | 5.252.190.129  
             | 5.252.190.56  
             | 5.252.190.186  
             | 5.252.190.32  
             | 5.252.190.34  
             | 45.148.120.161  
             | 45.148.120.113  
             | 4.227.193.241  
             | 27.115.124.45  
             | 197.231.197.11  
             | 185.7.33.149  
             | 185.213.175.253  
             | 180.163.220.66  
             | 102.129.143.22  |
| SHA256   | 0e05169d111415903a1098a10c34cdbc390c23016cd4e179dd9fef507104495  
             | 2413b5d0750c23b07999ec33a5b4930be224b661aaf290a0118db03f31acb5  
             | 34e435196dd795e1ec31169bd111c7ec964e5a6ab525a562b17f10de0ab031d  
             | 387cee556eaedfba8c114ed1c6b98d8b9b65e9f178cf2eae2f5ac441082747a  
             | 3a977446ed70b0264ef8fa3135d8b134c93e868a4cc0aa5d3c2a74545725b  
             | 3ab73ea9aebf271e5f3ed701286701d0be688bf7ad4fb276cb4fbe35c8af8409  
             | 4359ead4161b1b2df8ad9e53c497806403a2253b7e13c03317fc08a3db095bf  
             | 48367d94cb4411f15d7ef9c455c921253ad8121f2363c4de949ce1b615429a  
             | 5b566de1aa4b2179f579cdac6283b33e98f0c8c1afa6211a787f8156848d67ff  
             | 6015fed13c5510bbb89b0a5302c8b95a5b811982f6de9930725c4630ec4011d  
             | 702421bceee1785d93271d311f0203da34cc9936317e299575b06503945a6ea1e0  
             | 9d1723777de67bc7e11678db8002a32de3b6cd6c4a629cd165e3f7bbace8ead  
             | 9e89df045664960670a05610ea2b0ad4f7f502f73d84321fb07861348fdc24a  
             | a1269294254e958e058c0fe887ebc4201d5c266557f09c3f37542bd6d53d7  |
Mass Exploitation of MOVEit Transfer Critical Vulnerability