

SUBSCRIBE

SIGN IN

MY BOOK DEAD —

Hackers exploited 0-day, not 2018 bug, to mass-wipe My Book Live devices [Updated]

Western Digital removed code that would have prevented the wiping of petabytes of data.

DAN GOODIN - 6/30/2021, 12:01 AM



Getty Images

[Enlarge](#)

Update 6/29/2021, 9:00 PM: Western Digital has published an [update](#) that says the company will provide data recovery services starting early next month. My Book Live customers will also be eligible for a trade-in program so they can upgrade to My Cloud devices. A spokeswoman said the data recovery service will be free of charge.

The company also provided new technical details about the zeroday, which is now being tracked as [CVE-2021-35941](#). Company officials wrote:

We have heard concerns about the nature of this vulnerability and are sharing technical details to address these questions. We have determined that the unauthenticated factory reset vulnerability was introduced to the My Book Live in April of 2011 as part of a refactor of authentication logic in the device firmware. The refactor centralized the authentication logic into a single file, which is present on the device as `includes/component_config.php` and contains the authentication type required by each endpoint. In this refactor, the authentication logic in `system_factory_restore.php` was correctly disabled, but the appropriate authentication type of `ADMIN_AUTH_LAN_ALL` was not added to `component_config.php`, resulting in the vulnerability. The same refactor removed authentication logic from other files and correctly added the appropriate authentication type to the `component_config.php` file.

The post added:

We have reviewed log files which we have received from affected customers to understand and characterize the attack. The log files we reviewed show that the attackers directly connected to the affected My Book Live devices from a variety of IP addresses in different countries. Our investigation shows that in some cases, the same attacker exploited both vulnerabilities on the device, as evidenced by the source IP. The first vulnerability was exploited to install a malicious binary on the device, and the second vulnerability was later exploited to reset the device.

What follows is the article as it originally appeared:

Last week's mass-wiping of Western Digital My Book Live storage devices involved the exploitation of not just one vulnerability but also a second critical security bug that allowed hackers to remotely perform a factory reset without a password, an investigation shows.

The vulnerability is remarkable because it made it trivial to wipe what is likely petabytes of user data. More notable still was that, according to the vulnerable code itself, a Western Digital developer actively removed code that required a valid user password before allowing factory resets to proceed.

Done and undone

The undocumented vulnerability resided in a file aptly named `system_factory_restore`. It contains a PHP script that performs resets, allowing users to restore all default configurations and wipe all data stored on the devices.

Normally, and for good reason, factory resets require the person making the request to provide a user password. This authentication ensures that devices exposed to the Internet can only be reset by the legitimate owner and not by a malicious hacker.

Advertisement



As the [following script](#) shows, however, a Western Digital developer created five lines of code to password-protect the reset command. For unknown reasons, the authentication check was cancelled, or in developer parlance, it was commented out, as indicated by the double `/` character at the beginning of each line.

```
function post($urlPath, $queryParams = null, $outputFormat = 'xml') {  
    // if (!authenticateAsOwner($queryParams))  
    // {  
    //     header("HTTP/1.0 401 Unauthorized");  
    //     return;  
    // }
```

"The vendor commenting out the authentication in the system restore endpoint really doesn't make things look good for them," HD Moore, a security expert and the CEO of network discovery platform Rumble, told Ars. "It's like they intentionally enabled the bypass."

To exploit the vulnerability, the attacker would have had to know the format of the XML request that triggers the reset. That's "not quite as easy as hitting a random URL with a GET request, but [it's] not that far off, either," Moore said.

Dude, where's my data?

The discovery of the second exploit comes five days after people all over the world reported that their [My Book Live devices had been compromised](#) and then factory-reset so that all stored data was wiped. My Book Live is a book-sized storage device that uses an Ethernet jack to connect to home and office networks so that connected computers have access to the data on it. Authorized users can also access their files and make configuration changes over the Internet. Western Digital stopped supporting the My Book Live in 2015.

FURTHER READING

"I'm totally screwed." WD My Book Live users wake up to find their data deleted

Western Digital personnel [posted an advisory](#) following the mass wiping that said it resulted from attackers exploiting [CVE-2018-18472](#). The remote command execution vulnerability was [discovered in late 2018](#) by security researchers Paulos Yibelo and Daniel Eshetu. Because it came to light three years after Western Digital stopped supporting the My Book Live, the company never fixed it.

An analysis performed by Ars and Derek Abdine, CTO at security firm Censys, found that the devices hit by last week's mass hack had also been subjected to attacks that exploited the unauthorized reset vulnerability. The additional exploit is documented in log files extracted from two hacked devices.

One of the logs was [posted](#) in the Western Digital [support forum](#) where the mass compromise first came to light. It shows someone from the IP address 94.102.49.104 successfully restoring a device:

```
rest_api.log.1:Jun 23 15:46:11 MyBookLiveDuo REST_API[9529]: 94.102.49.104 PARAMETER System_factory_restore POST : erase = none  
rest_api.log.1:Jun 23 15:46:11 MyBookLiveDuo REST_API[9529]: 94.102.49.104 OUTPUT System_factory_restore POST SUCCESS
```

A [second log file](#) I obtained from a hacked My Book Live device showed a different IP address—23.154.177.131—exploiting the same vulnerability. Here are the telltale lines:

Jun 16 07:28:41 MyBookLive REST_API[28538]: 23.154.177.131 PARAMETER System_factory_restore POST : erase = format
Jun 16 07:28:42 MyBookLive REST_API[28538]: 23.154.177.131 OUTPUT System_factory_restore POST SUCCESS

After presenting these findings to Western Digital representatives, I received the following confirmation: “We can confirm that in at least some of the cases, the attackers exploited the command injection vulnerability (CVE-2018-18472), followed by the factory reset vulnerability. It’s not clear why the attackers exploited both vulnerabilities. We’ll request a CVE for the factory reset vulnerability and will update our bulletin to include this information.”

Advertisement

This vulnerability has been password-protected

The discovery raises a vexing question: if the hackers had already obtained full root access by exploiting CVE-2018-18472, what need did they have for this second security flaw? There’s no clear answer, but based on the evidence available, Abdine has come up with a plausible theory—that one hacker first exploited CVE-2018-18472 and a rival hacker later exploited the other vulnerability in an attempt to wrest control of those already compromised devices.

The attacker who exploited CVE-2018-18472 used the code execution capability it provided to modify a file in the My Book Live stack named `language_configuration.php`, which is where the vulnerability is located. According to a [recovered file](#), the modification added the following lines:

```
function put($urlPath, $queryParams=null, $outputFormat='xml'){  
  
    parse_str(file_get_contents("php://input"), $changes);  
  
    $langConfigObj = new LanguageConfiguration();  
    if(!isset($changes["submit"]) || sha1($changes["submit"]) != "56f650e16801d38f47bb0eeac39e21a8142d7da1")  
    {  
        die();  
    }  
}
```

The change prevented anyone from exploiting the vulnerability without the password that corresponds to the cryptographic SHA1 hash `56f650e16801d38f47bb0eeac39e21a8142d7da1`. It turns out that the password for this hash is `p$EFx3tQWoUbFc%B%R$k@`. The plaintext appears in the recovered log file [here](#).

A [separate modified `language_configuration.php` file](#) recovered from a hacked device used a different password that corresponds to the hash `05951edd7f05318019c4cfafab8e567afe7936d4`. The hackers used a third hash—`b18c3795fd377b51b7925b2b68ff818cc9115a47`—to password-protect a separate file named `accessDenied.php`. It was likely done as an insurance policy in the event that Western Digital released an update that patched `language_configuration`.

So far, attempts to crack these two other hashes haven’t succeeded.

According to Western Digital’s advisory linked above, some of the My Book Live devices hacked using CVE-2021-18472 were infected with malware called `.ntpd,1-ppc-be-t1-z`, which was written to run on the PowerPC hardware used by My Book Live devices. One user in the support forum [reported](#) a hacked My Book Live receiving [this malware](#), which [makes devices part of a botnet](#) called Linux.Ngioweb.

A theory emerges

So why would someone who successfully wrangled so many My Book Live devices into a botnet turn around and wipe and reset them? And why would someone use an undocumented authentication bypass when they already have root access?

The most likely answer is that the mass wipe and reset was performed by a different attacker, very possibly a rival who either attempted to take control of the rival’s botnet or simply wanted to sabotage it.

“As for motive for POSTing to this [system_factory_restore] endpoint on a mass scale, it is unknown, but it could be an attempt at a rival botnet operator to take over these devices or render them useless, or someone who wanted to otherwise disrupt the botnet which has likely been around for some time, since these issues have existed since 2015,” Abdine wrote in a [recent blog post](#).

The discovery of this second vulnerability means that My Book Live devices are even more insecure than most people thought. It adds authority to Western Digital’s recommendation to all users to disconnect their devices from the Internet. Anyone using one of these devices should heed the call immediately.

For many hacked users who lost years’ or decades’ worth of data, the thought of buying another Western Digital storage device is probably out of the question. Abdine, however, says that My Cloud Live devices, which replaced Western Digital’s My Book Live products, have a different codebase that doesn’t contain either of the vulnerabilities exploited in the recent mass wiping.

“I took a look at the My Cloud firmware, too,” he told me. “It’s rewritten and bears some, but mostly little, resemblance to My Book Live code. So it doesn’t share the same issues.”

Promoted Comments

broomstick / Smack-Fu Master, in training / [et Subscriptor](#)

JUMP TO POST

I was happier when the story was simply “We didn’t bother patching this thing from 2018”...


6 posts | registered 4/1/2020

DAN GOODIN

Dan is the Security Editor at Ars Technica, which he joined in 2012 after working for The Register, the Associated Press, Bloomberg News, and other publications. Find him on Mastodon at: <https://infosec.exchange/@dangoodin>

EMAIL dan.goodin@arstechnica.com


Advertisement




SITREP: F-16 replacement search a signal of F-35 fail?

SITREP: F-16 replacement search a signal of F-35 fail?


Footage courtesy of Dvids, Boeing, and The United States Navy.




SITREP: F-16 replacement search a signal of F-35 fail?



Sitrep: Boeing 707



Steve Burke of GamersNexus Reacts To Their Top 1000 Comments On YouTube



Scott Manley Reacts To His Top 1000 YouTube Comments

[+ More videos](#)

← PREVIOUS STORY

NEXT STORY →

Related Stories

Today on Ars

[STORE](#)
[SUBSCRIBE](#)
[ABOUT US](#)
[RSS FEEDS](#)
[VIEW MOBILE SITE](#)

[CONTACT US](#)
[STAFF](#)
[ADVERTISE WITH US](#)
[REPRINTS](#)

NEWSLETTER SIGNUP

Join the Ars Orbital Transmission mailing list to get weekly updates delivered to your inbox.

[SIGN ME UP →](#)