Closed Issue created 3 years ago by Andrew Bartlett

[SECURITY] LSIDLogin() is insecure and can allow user impersonation

`LSIDLogin()` has this header:

```
* This is all horribly insecure, but its hard not to be.
```

However it excels at that more than could ever be imagined.

If, say when LDAP authentication is used, and for new users, the usr->password is known (eg "" or *), then the validation used for the lsid cookie can be spoofed.

The reason is that

```
      list( $x, $salt, $y) = explode('*', $validation_string);
      $my_validation = session_salted_md5($usr->user_no . $usr->username . $usr->password, $salt);
```

contains only known or easily guessed inputs for this case. This is the case for user_no, as this is a simple number and DaviCal (for example) displays it in an internal page. Usernames are easily guessed and are likewise known to other users.

The salt should provide the security, but is is selected by the user, so the user can simply choose any value when computing their attack.

Finally, the whole design of the lsid cookie is flawed, there is no way to time out sessions, they last forever. This means that the logout relies on the browser being honest and destroying the cookie, there is no way to invalidate a cookie once issued, other than to change the password. Therefore if the cookie is stolen (eg JS XSS attack etc) then it can be re-used forever.

The fix should be to cut `LSIDLogin()` down to a no-op.

I'm more than happy to provide further explanation and to move this to a more appropriate forum if there is one.

Drag your designs here or click to upload.

Tasks 0

No tasks are currently assigned. Use tasks to break down this issue into smaller parts.

Linked items 0

Link issues together to show that they're related. Learn more.

Related merge requests 1

Fix param-filter/text-match evaluation (Issue #21)
!17 ✓

Activity

Andrew McMillan @karora · 3 years ago Maintainer

I'm pretty sure I wrote that comment - perhaps 15 years ago - but I've never used LDAP auth so the full wonder of the vulnerability would have been less obvious to me.

I agree that just nuking LSIDLogin() is very likely the best approach - by all means propose a patch :-)

Andrew Bartlett @abartlet · 3 years ago Author

Do you have a development repo for confidential security patches?

Jim Fenton @jimfenton · 3 years ago Owner

No, we don't have a separate repo for that. I suggest that you create a fork of the main repo, restrict access to it (but please allow access for karora, puck, fsfs, and myself). When the fix is ready, click the "create confidential merge request" above to create the merge request.

Note that on the security issues we just fixed, I was never able to get that button to work. Try it anyway, and yell to one of us (probably me) if we need to merge it manually.

Andrew Bartlett @abartlet · 2 years ago Author

I can't help developing this. I'm already on leave and won't be back in the office for another month.

This and !17 (merged) needs to be dealt with urgently. I can't imagine I'm the only one to notice this. At this stage treat me as a security researcher, not a development resource please.

Please also get a CVE from Red Hat's security team (secalert@redhat.com).

Finally, I don't seem to get notifications on this issue, so don't get replies unless I seek them directly.

Sorry!

Andrew Bartlett @abartlet · 2 years ago Author

@karora can you please find someone to look at this? My role here is security reporter, I don't have the time to also be a Davical developer. Sorry!

Andrew Bartlett @abartlet · 2 years ago Author

@jimfenton Perhaps you can help? Always happy to answer questions of course.

Florian Schlichting @fsfs · 2 years ago Owner

I went ahead and did as Andrew suggested, drop LSIDLogin() and related bits. See https://gitlab.com/fsfs/awl/-/commit/1cdaa34c26487b8dea5e8762cdd747fba2e32955 - comments welcome.

@jimfenton can you get a CVE again for this and the other issue, with which to annotate the commits?

Jim Fenton @jimfenton · 2 years ago Owner

CVE request has been submitted. It took a couple of weeks last time.

Jim Fenton @jimfenton · 2 years ago Owner

@abartlet I requested the CVEs from MITRE as I had done before. Is there some reason I should have gotten them from Red Hat? Just saw that you mentioned that above.

Andrew Bartlett @abartlet · 2 years ago Author

@jimfenton While as you note MITRE will provide a CVE number, their guide requests that you approach the more relevant numbering authority first (presumably for workload management). Red Hat is the numbering authority for 'Open Source' and returns them in around 24 hours to Samba.

Edited by Andrew Bartlett 2 years ago

Andrew Bartlett @abartlet · 2 years ago Author

BTW, you may be interested in the Samba Team's security process as a working example of how the Samba Team (of which I'm a part) does security releases: https://wiki.samba.org/index.php/Samba_Security_Process

I don't want to be pushy, just figured it might be a useful reference.

Jim Fenton @jimfenton · 2 years ago Owner

Thanks, @abartlet. I had seen Red Hat on the list of CNAs, but they're listed as "Linux issues only" and I interpreted that narrowly. Do you think it's worth withdrawing that request and going to Red Hat?

By the way, I presume it's OK to credit you as discoverer?

Thanks for the pointer to the Samba Team's process -- very helpful.

Andrew Bartlett @abartlet · 2 years ago Author

@jimfenton In terms of Red Hat vs Mitre, it seems that has changed. Perhaps they got overwhelmed :-).

I certainly wouldn't wait 2 weeks for one however, and I think "Linux issues" is meant to mean "things found in a mainstream Linux distribution" (which DAViCal is) given how much the rest of the Open Source world has exploded.

Yes, please credit me as "Andrew Bartlett, Catalyst"

Edited by Andrew Bartlett 2 years ago

Jim Fenton @jimfenton · 2 years ago

Owner

This has been assigned CVE-2020-11729.

Florian Schlichting made the issue visible to everyone 2 years ago

Florian Schlichting closed via commit 535505c9 2 years ago

Please register or sign in to reply