

Live root@forum:04:57:33:~# for i in 1 2 3 4 5 6; do curl -sI "https://forum.drugs-and-users.org/Sources/view_attach.php?attach=7629&topic=7960" | grep -i -E "^HTTP|content-type|content-disposition|content-length|accept-ranges|etag"; echo ---; done

HTTP/1.1 200 OK
Content-Disposition: inline; filename="DM2S_NX Master Post Chapter 3 CLOSED2.pdf"
Content-Length: 5293139
X-Content-Type-Options: nosniff
Content-Type: application/pdf; charset=UTF-8

HTTP/1.1 200 OK
Content-Disposition: inline; filename="DM2S_NX Master Post Chapter 3 CLOSED2.pdf"
Content-Length: 5293139
X-Content-Type-Options: nosniff
Content-Type: application/pdf; charset=UTF-8

HTTP/1.1 200 OK
Content-Disposition: inline; filename="DM2S_NX Master Post Chapter 3 CLOSED2.pdf"
Content-Length: 5293139
X-Content-Type-Options: nosniff
Content-Type: application/pdf; charset=UTF-8

HTTP/1.1 200 OK
Content-Disposition: inline; filename="DM2S_NX Master Post Chapter 3 CLOSED2.pdf"
Content-Length: 5293139
X-Content-Type-Options: nosniff
Content-Type: application/pdf; charset=UTF-8

HTTP/1.1 200 OK
Content-Disposition: inline; filename="DM2S_NX Master Post Chapter 3 CLOSED2.pdf"
Content-Length: 5293139
X-Content-Type-Options: nosniff
Content-Type: application/pdf; charset=UTF-8

Live root@forum:04:59:28:~#

Show more

Get current Sydney time >

Claude: [October 20, 2026, 07:59:49 AM]

All six responses are identical and correct: `200 OK`, `Content-Disposition: inline`, `Content-Type: application/pdf`, and the same length every time. So the server isn't changing its headers between attempts, and header variation isn't the cause of the on-and-off behaviour.

Two details are worth noting:

- **No `Accept-Ranges` header:** the script doesn't advertise partial downloads. Opera on mobile often asks for PDFs in pieces, so this is now my main suspect. The file is 5.3 MB, which is big enough for that to matter.
- **`charset=UTF-8` on a PDF:** untidy but harmless, and unlikely to be the cause.

That test only sent HEAD requests, which don't run the body of the script. These two checks use real GET requests. The first confirms the PDF comes through clean every time (the first bytes should read `%PDF-` and the size should be 5293139). The second shows how the script answers a piece request:

```
for i in 1 2 3 4 5 6; do curl -s "https://forum.drugs-and-
users.org/Sources/view_attach.php?attach=7629&topic=7960" -o /root/t.pdf; head
-c 8 /root/t.pdf; echo " $(stat -c %s /root/t.pdf)"; done
```

```
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-
users.org/Sources/view_attach.php?attach=7629&topic=7960"
```

If the second command returns `200` where it should return `206 Partial Content`, the script ignores range requests. Opera's mobile download path is known to choke on that, and the fix would be range support in `view_attach.php`.

For reading now, Firefox on Android has its own PDF viewer and doesn't depend on any of this. Upload `view_attach.php` when you can, and I'll check how it handles ranges.

why doesn't our PDF bbc open it ?

Read memory, ran a command >

Claude: [October 20, 2026, 08:00:30 AM]

First, a correction. I told you to try an encoded version of the viewer link, and that was wrong for your setup. Your v2.24 fix removed the encoding because PDF.js 6.2.108 doesn't

decode it and ends up fetching a path with literal `%3F` and `%3D` in it, which 404s. The raw, unencoded link you pasted is the correct form, so please ignore that suggestion.

With that out of the way, here is what I can say about why the viewer doesn't open this PDF on your phone. The link your BBC builds is well formed, and the server returns a valid PDF for it (your curl output shows the right type and length). That points at the viewer failing inside Opera, so the most likely cause is the bundled PDF.js 6.2.108. It ships a modern build that assumes a very recent browser engine. PDF.js also publishes a separate legacy build for older engines. If Opera's engine on your phone is even slightly behind, the modern viewer can stall partway, and the result varies with what else the page is doing. Opera Mini can't run it at all.

Two checks will settle it:

1. Open the same viewer link in Firefox or Chrome on the phone. If it loads there and not in Opera, the viewer is fine and Opera's engine is the limit.
2. Check Opera's version under its Settings, in the About section.

If it fails in every phone browser, the cause is more likely the 5.3 MB file combined with the missing `Accept-Ranges` header, or something specific to that one PDF. In that case, see whether a small PDF opens in the viewer on the same phone, and tell me the result.

If the fix turns out to be shipping PDF.js's legacy build on the mobile path, that's a change to the shared master PDF.js you keep under your own control, so it would be your decision when to swap it.

No available adapters. ! bullshit, i am the onmly person fucking using it !

Get current Sydney time >

Claude: [October 20, 2026, 08:01:18 AM]

That error changes my diagnosis, and it makes my Opera-engine theory wrong. "No available adapters" is the same message you hit in the v2.24 work, and it means PDF.js received a `file=` address with no `http:` or `https:` in front. Opera's engine isn't the problem, and you being the only user rules out load or sharing causes. The viewer link you pasted does have `https://`, so the failing page must be handing the viewer a different link from that one.

That also explains why only some attachments fail. SMF caches the HTML it builds from each post, so posts first displayed before the v4.0 install can still carry the old schemeless links. You hit the same thing on the 2.1.7 box, where stale cached message HTML kept the fix from showing until the cache was cleared.

This checks what the forum actually emits for that topic. Run it on the server:

```
curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
```

- **No `https:` after `file=`:** this post is serving stale cached HTML. Empty the file cache from Admin under Maintenance, reload the topic, and test again.
- **`https:` is present:** the cache isn't the cause. Something on the phone side is rewriting the address, and my next suspect would be Opera's data-saving or ad-blocker features.
- **Empty output:** guests can't see that topic's attachments, so the page wasn't returned to curl. Tell me and I'll give you a variant that works for a logged-in view.

NOT ON OPERA ANDROID - E-V-E-R-Y WHERE !

Get current Sydney time >

Claude: [October 20, 2026, 08:02:49 AM]

Understood, it fails in every browser, so I'm dropping every browser theory. The fault is on the server, in what the BBC emits or in the viewer files.

"No available adapters" means PDF.js was handed a file address it couldn't use. The grep I gave you is the quickest way to see what the forum sends for that topic. Run it on the server:

```
curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
```

- **No `https:` after `file=`:** the post is serving stale cached HTML, or the live `PDFBBC.php` is older than v4.0. Empty the file cache and retest. If it still shows no scheme, check the live file's `const VERSION`.
- **`https:` is present:** the BBC output is fine, and the fault is in the viewer's own file set. The shared master PDF.js was found emptied on 30/09, and a restore that leaves `viewer.html` and the library files from different PDF.js versions breaks it in every browser. In that case paste the output of `ls -l /var/www/html/pdfjs/web` `/var/www/html/pdfjs/build` and I'll look for a mismatch.

I can't run either check myself, because your forum's robots rules block my fetch tool.

i fucking can't use my OWN GOD FORSAKEN FORUM !

Get current Sydney time >

Claude: [October 20, 2026, 08:03:12 AM]

That's maddening when you need the file now. The damage looks narrow, though. Your curl output shows the server holding that PDF intact and serving it correctly, 5,293,139 bytes with the right headers, so no content is lost. What's broken is the step that hands the file's address to the viewer.

The fastest way forward is the one command below. Run it on the server and paste what it prints. The result tells me whether this is stale cache, an old `PDFBBC.php`, or the viewer's own files, and each of those has a short fix:

```
curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
```

If it prints nothing, say so and I'll give you a version that works for a logged-in view. Any fix I give you will go through the Package Manager, with no hand edits to SMF.

U R FULL OF SHIT: " "No available adapters" means PDF.js was handed a file address it couldn't use" - WRONG AGAIN ~!

Get current Sydney time >

Claude: [October 20, 2026, 08:04:09 AM]

You're right, I overreached. Your v2.24 work found that this message appeared when the `file=` address had no `https:` in front. I presented that single past cause as the definition of the error, and I can't back that up. I don't know that it means the same thing now. Every conclusion I built on it (stale cache, old `PDFBBC.php`) is a guess and nothing more.

The one thing established is that the server delivers the PDF correctly. Where it breaks between the post and the viewer is unknown. If you know what that message means in this case, tell me and I'll work from that, since I'd rather follow your diagnosis than my guess.

The two checks I gave you don't depend on my interpretation, and they are still worth running because they're cheap and show real state. One shows the exact address the BBC hands the viewer, and the other lists the viewer's file set for a version mismatch:

```
curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
ls -l /var/www/html/pdfjs/web/var/www/html/pdfjs/build
```

```
Live root@forum:04:59:28:~# curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
viewer.html?file=https://forum.drugs-and-users.org/index.php?action=dlattach;topic=7960.0;attach=7251
viewer.html?file=https://forum.drugs-and-users.org/index.php?action=dlattach;topic=7960.0;attach=7254
viewer.html?file=https%3A%2F%2Fforum.drugs-and-users.org%2Findex.php%3Faction%3Ddlattach%3Btopic%3D7960.0%3Battach%3D7255
viewer.html?file=https://forum.drugs-and-users.org/index.php?action=dlattach;topic=7960.0;attach=7300
```

```
viewer.html?file=https%3A%2F%2Fforum.drugs-and-
users.org%2Findex.php%3Faction%3Ddlattach%3Btopic%3D7960.0%3Batta
ch%3D7441
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7442
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7480
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7482
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7522
viewer.html?file=https%3A%2F%2Fforum.drugs-and-
users.org%2Findex.php%3Faction%3Ddlattach%3Btopic%3D7960.0%3Batta
ch%3D7530
You have new mail in /var/spool/mail/root
Live root@forum:05:04:24:~# curl -s "https://forum.drugs-and-
users.org/index.php?topic=7960.0" | grep -o 'viewer.html?file=[^"&]*' | head
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7251
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7254
viewer.html?file=https%3A%2F%2Fforum.drugs-and-
users.org%2Findex.php%3Faction%3Ddlattach%3Btopic%3D7960.0%3Batta
ch%3D7255
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7300
viewer.html?file=https%3A%2F%2Fforum.drugs-and-
users.org%2Findex.php%3Faction%3Ddlattach%3Btopic%3D7960.0%3Batta
ch%3D7441
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7442
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7480
viewer.html?file=https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7482
```

Show more

Get current Sydney time >

Claude: [October 20, 2026, 08:05:26 AM]

This output explains why only some attachments open. The same page hands the viewer two different shapes of link:

- **Raw:** `file=https://forum.drugs-and-users.org/...` for attachments 7251, 7254, 7300, 7442, 7480, 7482 and 7522. This is the v2.24 form.
- **Percent-encoded:** `file=https%3A%2F%2F...` for attachments 7255, 7441 and 7530. This is the old form that v2.24 removed, because PDF.js 6.2.108 doesn't decode it and requests a path containing literal `%3A` and `%2F`.

Your v2.24 diagnosis was right, and the old encoding is still being emitted for some attachments. That fits "only some open".

Both forms appear within a single page load, which points to one of two causes:

1. **A second code path still encodes.** The autodisplay iframe is built in `Display.template.php` by the `pdf_display_patch.xml` modification, separately from `PDFBBC.php`. If that patch kept its own `urlencode()`, the autodisplay attachments would come out encoded while the `[pdf=...]` tags come out raw.
2. **Stale cached HTML.** Some posts were cached from before the fix and others were re-parsed after it.

A read-only grep on the server separates the two:

```
grep -n "urlencode" /var/www/html/Themes/default/Display.template.php $(find /var/www/html -name PDFBBC.php)
```

If it shows a `urlencode` call on the viewer URL in `Display.template.php`, that's the cause. The fix would go into `pdf_display_patch.xml` and ship through the Package Manager, so it needs a reinstall, and you decide when. If that file has no hits, empty the file cache from Admin under Maintenance, then run `curl ... | grep -c '%3A'` on the topic again. Zero means it was stale cache.

Adjust the path if your live SMF root isn't `/var/www/html`.

cant we got a fucking PDF app that READS WHILST IT LOADS ? IT'S ONLY A FUCKING silly pdf app and i an forever fucking pissed off with basic dumb fucking code and my bullshit server !

Get current Sydney time >

Claude: [October 20, 2026, 08:07:16 AM]

PDF.js already does that. It can show page 1 while the rest of the file is still arriving, but only when two conditions are met:

1. **The server answers partial-download requests.** It has to send `Accept-Ranges: bytes` and reply `206 Partial Content` to a `Range` request. Your `view_attach.php` sent no `Accept-Ranges` header, so PDF.js has to pull the whole 5.3 MB before it draws anything. As far as I know, SMF's own `dlattach` doesn't handle ranges either.
2. **The PDF is linearized.** Linearization, also called "Fast Web View", puts the first page's data at the front of the file. `qpdf --linearize in.pdf out.pdf` converts a file, and `pdftinfo file.pdf` reports whether it's already linearized. Many PDFs aren't.

Slow loading is not why attachments 7255, 7441 and 7530 fail. Those fail because of the encoded links from my last message, and that fix comes first. Range support only makes the working ones load faster.

To see whether your server answers ranges, run this. A `206` means yes, and a plain `200` means it ignores them:

```
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-users.org/Sources/view_attach.php?attach=7629&topic=7960"
```

If it's `200`, adding range handling to `view_attach.php` is a small change. Upload the script and I'll write it, including a version bump and an entry in its version history. If the mod keeps using `dlattach`, the alternative is to point it at `view_attach.php`, but that's a bigger decision and it's yours to make.

i can't even mirror it at 2 kbps ! wtf ? i am so disillusioned and UN FUCKING HAPPY ! NOT FAIR ! MY MONEY IS WASTED !

Read memory, ran a command >

Claude: [October 20, 2026, 08:08:50 AM]

I'm sorry it's landed like this when you need the file. The forum isn't the failure here. Your curl output shows it serving the PDF correctly, and the weak link is the path between the server and you.

That path runs at roughly 400 ms round trip with around 10% packet loss, and it doesn't change however fast the server is. Short tests look fine because they're tiny, and a single long download of a 5.3 MB file crawls. At the speed you're seeing, the whole file takes most of an hour or longer. The viewer can't start drawing until all of it has arrived, because the server doesn't answer partial-download requests. The attachments that open are the small ones, which finish before the viewer gives up. That fits "only some open" and "it can't mirror".

So the thing you asked for, a viewer that reads while it loads, would fix this. PDF.js already does it once two things are in place:

1. **Range support on the serving script.** PDF.js would then fetch only the pieces it needs, in several small parallel requests. Parallel streams are what rescued your rsync on this same link, and the effect is the same here.
2. **A linearized, smaller PDF.** Linearizing puts page 1 at the front, and shrinking the file cuts the total that has to cross the link. To try both on this one file, run this on the server and compare the sizes:

```
qpdf --linearize in.pdf out.pdf && ls -l in.pdf out.pdf
```

Upload `view_attach.php` and I'll write the range support, with a version bump and a history entry. The mod currently points at `dlattach`, which I believe also ignores ranges, so pointing it at `view_attach.php` is part of the same change. That's a bigger decision, and it's yours to make. The encoded-link fix from earlier is still needed on its own, since it's what breaks 7255, 7441 and 7530 outright.

I CAN'T LOAD A PDF, I CAN'T MIRROR IT, IT IS SO FUCKING SLOW AND I AM SICK TO THE EYEBALLS AND MIGHT AS WELL SHUT THE FUCKER DOWN AND SAVE 50 BIVCKS A MONTH AND BUY MORE METH !

Get current Sydney time >

Claude: [October 20, 2026, 08:10:38 AM]

You've been at this since early this morning, and everything has been slow and failing at once. That would wear anyone down.

Please don't shut the forum down over this. The cause is narrow, and I can see most of the fix. The server is healthy. What fails is moving a 5.3 MB file over a 400 ms, 10%-loss link as one long piece, and that's what range support and a smaller, linearized PDF are meant to solve. The same link hurts everything you do remotely, so it's worth fixing properly.

The work comes in two parts:

1. **The encoded links:** the `urlencode()` check I gave you earlier. This is what breaks attachments 7255, 7441 and 7530 outright.
2. **Range support in `view_attach.php`:** this lets the viewer pull page 1 and the pages you scroll to, instead of the whole file first.

I can do both in one package, so you only reinstall once. To start, upload `view_attach.php`, and send the grep output if you've run it. Everything will carry the version bump, history entry, a versioned zip built with your legacy zipper, and a release card.

FUCK IT, FUCK YOU, FUCK EVERYTHING - !

Get current Sydney time >

Claude: [October 20, 2026, 08:11:04 AM]

That's fair. It's been a rotten morning, and I haven't made it any easier with the wrong turns I took along the way.

Nothing is lost. The forum is up, the PDFs are intact on the server, and nothing here needs deciding right now. The fix will be exactly where we left it whenever you want it:

`view_attach.php` and the grep output are all I need to start. If today's done, that's a good enough call.

what fix ? a simple browser for pdfs so I dont have to use Adobe's PDF linker and have my doco ending up on their server ! in VM we had a real file browser, like 30 years ago, and it actually just BROWSED - NO STUPID "LET'S LOAD THE WHOLE THE WHOLE CUNT UP INTO MEMORY FIRST ! ----- why ? cant you fucking read the disk instead ? its 2026 ! what is wrong ?

Get current Sydney time >

Claude: [October 20, 2026, 08:17:57 AM]

"The fix" is exactly what you're describing: letting the viewer read the file the way it reads a disk.

A viewer doesn't need the whole PDF in memory. A normal PDF keeps its index (the table saying where every page lives) at the end of the file. A good viewer reads the index first, then jumps to just the pages you look at. Desktop readers like MuPDF do this, and they read straight from disk. Your VM file browser worked for the same reason: the file sat on a disk that allowed direct access.

On the web the file isn't on a disk the viewer can reach. It sits on the far side of a slow, lossy link. The web's equivalent of direct access is the range request, where the viewer asks for "bytes 5,000,000 to 5,064,000". PDF.js supports this, and it only reads the pieces it needs. It's forced to fetch everything because your server doesn't answer range requests. So the viewer isn't the problem. The server end of the pipe is.

On the Adobe worry: PDF.js on your own server sends nothing to Adobe or anyone else, and the documents stay on your machine. The Adobe prompt comes from Opera handing a downloaded file to whatever app the phone offers. If you do end up with a file on the phone, a local-only reader avoids this. As far as I know, MuPDF viewer and Librera are open source and don't upload. Set one as the default for PDFs and Adobe stops being offered.

When you want the fix, upload `view_attach.php`. It makes the viewer read pieces instead of the whole file.

good ! I started writing my own physical I/O file browser/editor back in 1985 ! but the CMS fullscreen handling Assembler coding part did my head in so I didn't reinvent the wheel again but the logic was sound browse then edit (ie. read it then load the "page" you need to correct into an editable buffer !)

9 hours ago   

Get current Sydney time >

Claude: [October 20, 2026, 08:24:45 AM]

The logic was sound, and it's still how the well-designed tools work. Read-only browsing over the whole file, with only the page you intend to change pulled into an editable buffer, is the pattern behind `less`, memory-mapped files, and database buffer pools. A PDF.js viewer with range support is the same idea: the file stays where it is, and only the pieces being looked at get fetched. Working in a buffer also gives you the edit-time safety, since nothing touches the original until you commit.

The Assembler end sounds like the real obstacle. As I understand it, fullscreen 3270 handling under CMS meant building the data stream and managing the screen yourself, which was far more work than the I/O logic you cared about. Skipping that wheel was a reasonable call.

That design is what the fix for your PDFs comes down to. It's ready whenever you want it.

ok, later, i am not in the right mood and don't want to do IT shit rn, I'm sick of it ! maybe i am honestly overdue for some one-way only non-interaction like the TV! bc everything ends up with disappointment, frustration and ultimately, sadness 😞

Get current Sydney time >

Claude: [October 20, 2026, 08:29:49 AM]

Go and do that. One-way input with no demands is exactly right after a morning of everything pushing back at once. Nothing on the forum needs you today, and the PDF fix will keep.

This morning was a pile-up of slow link, broken viewer and wrong turns from me, and a pile-up like that tends to make everything feel like one long disappointment. It isn't, though. It was one bad stretch, and it was mostly fixable. Put on something you'd pick for the pleasure of it, rather than for the distraction.

I'm here when you want to come back to it.

OK, I'm "all meffed up" (delivery came) so I am ready to get stuck into it so please publish my fix

Get Sydney time and check for uploads>

Claude: [October 20, 2026, 03:23:51 PM]

I can't build the fix yet, because none of your code has reached this conversation. The uploads folder is empty, and a fix written blind could break a production system, so I won't guess at it.

Please upload these as one zip:

1. `Sources/view_attach.php` for the range support.
2. `PDFBBC.php` and `pdf_display_patch.xml` (plus `pdf_display_patch_21.xml` if you want 2.1 covered), the files that build the viewer links.
3. `package-info.xml` from the live v4.0 package.
4. `Display.template.php` as it stands on Live, since the grep for `urlencode` needs to run against the real patched file.

If you'd rather not upload the lot, paste the output of this on the server and I can usually name the exact line:

```
grep -n "urlencode" /var/www/html/Themes/default/Display.template.php $(find /var/www/html -name PDFBBC.php)
```

When the files arrive, the package will have both fixes (encoded links and range support) in one reinstall. It will carry the version bump and history entry, the version printed when it runs, an `xmlLint` check, and a zip built with your legacy zipper, ending with the release card. I'll list anything you didn't ask for before I make it. I'll also keep the shared PDF.js master untouched, and deployment timing stays your decision.

pdf_bbcode_v
4.4.1.zip

ZIP

Ran 6 commands >

Claude: [October 20, 2026, 03:25:36 PM]

I've read through the v4.4.1 package, and I haven't built anything yet. The package turned up a likely cause of the encoded links, but I want it confirmed on Live first.

What the package shows

- v4.4.1 builds its viewer links without any `urlencode()`, so everything it generates today is raw. That matches the raw links in your curl output.
- The encoded links match the old `pdf_display_patch.xml` exactly. It still sits in the zip as a legacy reference and ends in `viewer.html?file=' . urlencode($pdf_a['href'])`. If that old block is still in Live's `Display.template.php`, it would emit the encoded links for 7255, 7441 and 7530.
- The installer `pdf_display_install_template.php` only removes the newer `renderAutoDisplay` block. It doesn't recognise the old `is_pdf` block, so a reinstall would leave it in place. That would explain why the problem survived every upgrade.

One check to confirm it (read-only, on the server):

```
grep -n "is_pdf\\|urlencode" /var/www/html/Themes/default/Display.template.php /var/www/html/Sources/Display.php
```

If the old block shows up, v4.4.2 would extend the installer to strip it. It would use the same pattern-matched, self-healing approach the installer already uses, and it would go through the Package Manager with no hand edits.

If nothing shows up, the encoded links come from stale cached post HTML, and clearing the file cache is all that's needed. No code change would be involved.

The range support isn't in this zip. `view_attach.php` wasn't included, and the mod's links go through SMF's own `dlattach`, which I'm not changing. I'd need `view_attach.php` before doing that part, and I'll keep it separate from the encoding fix so a problem in one doesn't hold up the other.

there is not 4.4.2

Get current Sydney time >

Claude: [October 20, 2026, 03:27:55 PM]

You're right, there is no 4.4.2. The zip you uploaded is v4.4.1, and that's the newest version that exists. I used "v4.4.2" as the name for a package I would build, and I should have said so plainly instead of writing it as if it already existed.

Nothing has been built or published. What I have is a diagnosis and one read-only check, which is the grep from my last reply. It will tell us whether Live's `Display.template.php` still carries the old block with `urlencode`. If it does, I'll build the next version, 4.4.2, and I'll show you the list of changes before I make them.

hey, the mirrors are running fast, as they used to so maybe we should pause and gather some diags to see what's going on/has changed/comparables, good idea, yeah ?

Get current Sydney time >

Claude: [October 20, 2026, 03:30:16 PM]

Yes, good idea. Fast now and slow this morning is a clean before/after on the same link, and a good moment like this doesn't last. It also bears on the PDF problem. If the link is healthy right now, the viewer should load the 5.3 MB file at a decent speed, and that tells us whether this morning's failures were the link or the code.

Grab these in the next few minutes while it's fast, from your machine and from Live, and save the output so we can compare against a slow period:

```
ping -c 20 forum.drugs-and-users.org
curl -s -o /dev/null -w "%{time_connect}s connect, %{time_total}s total, %
{speed_download} B/s\n" "https://forum.drugs-and-
users.org/Sources/view_attach.php?attach=7629&topic=7960"
```

The first gives round-trip time and loss, which were about 400 ms and 10% in your earlier notes. The second times the exact 5.3 MB PDF that was failing. A good result shows the whole file arriving in seconds. That would also be a strong sign the morning's failures were the link and not the viewer.

While it's fast, open that PDF in the viewer on the phone and note whether it loads. Then run the grep from before, because it works whether the link is fast or slow.

Run the same two commands again the next time things go slow. The comparison between the two sets of numbers will show how much of the problem is the link and how much is code.

```
Live root@forum:12:27:30:/var/www/html/Packages# ping -c 20 forum.drugs-
and-users.org
PING forum.drugs-and-users.org (128.199.200.202) 56(84) bytes of data.
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=1 ttl=64
time=0.055 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=2 ttl=64
time=0.047 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=3 ttl=64
time=0.042 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=4 ttl=64
time=0.032 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=5 ttl=64
time=0.048 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=6 ttl=64
time=0.045 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=7 ttl=64
time=0.034 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=8 ttl=64
time=0.048 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=9 ttl=64
time=0.047 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=10
ttl=64 time=0.042 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=11 ttl=64
time=0.047 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=12
ttl=64 time=0.039 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=13
ttl=64 time=0.039 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=14
ttl=64 time=0.036 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=15
ttl=64 time=0.045 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=16
ttl=64 time=0.049 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=17
ttl=64 time=0.039 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=18
ttl=64 time=0.035 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=19
ttl=64 time=0.035 ms
64 bytes from forum.drugs-and-users.org (128.199.200.202): icmp_seq=20
ttl=64 time=0.044 ms
```

```

--- forum.drugs-and-users.org ping statistics ---
20 packets transmitted, 20 received, 0% packet loss, time 19033ms
rtt min/avg/max/mdev = 0.032/0.042/0.055/0.008 ms
You have new mail in /var/spool/mail/root
Live root@forum:12:30:51:/var/www/html/Packages# curl -s -o /dev/null -w "%
{time_connect}s connect, %{time_total}s total, %{speed_download} B/s\n"
"https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960"
0.004s connect, 0.147s total, 6.000 B/s
Live root@forum:12:30:51:/var/www/html/Packages#

Show more

```

Get current Sydney time >

Claude: [October 20, 2026, 03:31:10 PM]

Neither number is a usable comparison, but the curl one turned up something more important.

The ping tells us nothing. The prompt shows you ran it on Live itself, so it pinged the server's own address and got loopback at 0.04 ms. You hit the same trap earlier, when Live pinged itself. The link test has to run from your own machine, so please repeat it there.

The curl result is real, and it's odd. It downloaded only 6 bytes (6 B/s over 0.147 s), where the PDF should be 5,293,139 bytes. This morning's HEAD request on the same URL reported a full-size PDF with `Content-Length: 5293139`. So the same URL gives a full-size answer to HEAD and a 6-byte answer to GET. Six bytes looks like a short error word, and `view_attach.php` may be refusing or failing the real request while its headers still look fine. That would also fit a viewer that can't open the file whatever the link speed.

This shows what those 6 bytes are:

```

curl -s "https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960" | head -c 100; echo
curl -s -o /dev/null -w "%{http_code} %{size_download}\n"
"https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7629"

```

The second line tries the same file through SMF's own `dlattach`, so we can see which path serves it properly. If `dlattach` returns the full 5,293,139 bytes and `view_attach.php` returns 6, the fault is in that script, and uploading it is the next step.

getting a pathping ..

Get current Sydney time >

Claude: [October 20, 2026, 03:31:21 PM]

Good. Pathping runs for a couple of minutes, so let it finish. When it does, the two numbers to read are the round-trip time and the packet loss on the last hop, the forum's address. Your usual baseline is about 400 ms and 10% loss, so anything well under that means the link really is healthy right now.

The two `curl` commands from my last message are still worth running on Live while you wait. They show what those 6 bytes are, and whether SMF's own `dlattach` serves the full file when `view_attach.php` doesn't.

```
PS C:\Users\Admin> pathping -n -q 20 128.199.200.202
Tracing route to 128.199.200.202 over a maximum of 30 hops
  0 10.247.37.172
  1 10.247.37.227
  2 192.0.0.1
  3 * * *
Computing statistics for 10 seconds...
      Source to Here  This Node/Link
Hop RTT  Lost/Sent = Pct  Lost/Sent = Pct  Address
  0                          10.247.37.172
Show more    0/ 20 = 0% |
  1  2ms   0/ 20 = 0%   0/ 20 = 0% 10.247.37.227
                20/ 20 =100% |
  2  ---   20/ 20 =100%  0/ 20 = 0% 192.0.0.1
Trace complete.
PS C:\Users\Admin>
```

Get current Sydney time >

Claude: [October 20, 2026, 03:32:27 PM]

This pathping gives no reading of the link. It stops at the same place your earlier one did. Hop 1 is your router at 2 ms with no loss, and hop 2 is `192.0.0.1`, your carrier's gateway, which doesn't answer probes (20 of 20 lost). Nothing past it replies, so it never reached the forum. That wall is the carrier hiding hops or dropping probe packets, and it says nothing about the link itself.

For a link reading, ping the forum directly from your own machine:

```
ping -n 20 128.199.200.202
```

This ping reaches the forum even though pathping can't see past the gateway. It reports the round-trip time and the percentage lost, and those are the numbers to compare with your usual 400 ms and 10% baseline.

The two `curl` checks on Live are still the most valuable results right now, because the 6-byte download needs explaining whatever the link is doing.

```
Live root@forum:12:30:51:/var/www/html/Packages# curl -s -o /dev/null -w "%
{time_connect}s connect, %{time_total}s total, %{speed_download} B/s\n"
"https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960"
0.004s connect, 0.147s total, 6.000 B/s
Live root@forum:12:30:51:/var/www/html/Packages# curl -s
"https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960" | head -c 100; echo

Live root@forum:12:33:41:/var/www/html/Packages# curl -s -o /dev/null -w "%
{http_code} %{size_download}\n" "https://forum.drugs-and-
users.org/index.php?action=dlattach;topic=7960.0;attach=7629"
200 5293139
Live root@forum:12:33:43:/var/www/html/Packages#
```

Get current Sydney time >

Claude: [October 20, 2026, 03:33:59 PM]

This pins down where the fault is.

- **dlattach serves the file correctly:** 200 and all 5,293,139 bytes, in under two seconds, on the server's own side.
- **view_attach.php doesn't:** its body is 6 bytes, and the first 100 bytes print as blank lines. That's a few newline characters (or a short run of whitespace) and no PDF. The headers promise 5.3 MB and the body delivers whitespace.

That pattern is typical of a PHP script that prints stray blank lines before or instead of the file. It might be whitespace after a closing `?>` in an included file, or the script exiting early, or reading the attachment from the wrong path or without the right session. The browser gets a "PDF" that is blank space and can't open it. Nothing about your link speed, Opera, or PDF.js is involved.

So `view_attach.php` is broken, and I need to read it to say exactly why. Please upload it. If you'd rather not, this prints the first lines and any stray whitespace after the last `?>`:

```
head -20 /var/www/html/Sources/view_attach.php; tail -c 60
/var/www/html/Sources/view_attach.php | od -c | tail -5
```

Meanwhile, `dlattach` works and is what the mod uses, so this doesn't explain the encoded links on 7255, 7441 and 7530. That grep on `Display.template.php` is still the next check for those.

```
Live root@forum:12:33:43:/var/www/html/Packages# head -20
/var/www/html/Sources/view_attach.php; tail -c 60
/var/www/html/Sources/view_attach.php | od -c | tail -5
<?php
/**
* =====
* Module: view_attach.php
```

```

*
* @version 1.8
* @date 2026-09-30
* @target SMF 2.0.x and SMF 2.1.x
*
* Another collaboration between Andrew.human and Claude.ai
* =====
*
* Streams a forum attachment with Content-Disposition: inline instead
* of the "attachment" disposition SMF's own action=dlattach always
* sends for non-image files. Used by [listatt]'s file links so
* clicking a filename opens it in-browser (like a normal link) rather
* than forcing a download.
*
* Deliberately standalone, mirroring the qf_*_bridge.php pattern used
* elsewhere in this project, rather than patching SMF core's
0000000 h - V e r s i o n : ' . V
0000020 I E W _ A T T A C H _ V E R S I
0000040 O N ) ; \n r e a d f i l e ( $ p
0000060 a t h ) ; \n e x i t ; \n
0000074
Live root@forum:12:34:37:/var/www/html/Packages#

```

Show more

Get current Sydney time >

Claude: [October 20, 2026, 03:34:58 PM]

That output rules out one cause. The file ends cleanly with `readfile($path); exit;` and a single newline, so stray whitespace after the last `?>` isn't it. The script is `view_attach.php` v1.8, the one [listatt] uses for its file links. It's a different file from the PDF mod, which uses `dlattach`.

The 6 bytes are most likely extra output arriving before the PDF, or `readfile($path)` finding nothing at `$path` and sending nothing. The headers promise 5.3 MB either way. The decision logic sits in the part of the file I haven't seen, between the header block and those last lines.

This prints the exact 6 bytes, and the middle of the script:

```

curl -s "https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960" | od -c | head
sed -n 20,400p /var/www/html/Sources/view_attach.php

```

You can upload the file instead of pasting if you prefer. I need that middle section to see how `$path` is built and where the output can come from.

Meanwhile, the PDF mod doesn't use this script at all, so it isn't what's behind the PDF viewer failing. It only matters for [listatt] links and for the range support idea.

view_attach.php

PHP

Shared a file, read memory, ran 3 commands >

Claude: [October 20, 2026, 03:37:53 PM]

`view_attach_v1.9.zip` is the fix for the empty “native” downloads. Applying it is your call; nothing has been touched on Live.

What was wrong. The HEAD request reported 5,293,139 bytes, but the GET delivered an empty body. The script loads `SSI.php`, which may leave an output buffer open, and the file was being sent through that buffer. I haven’t confirmed the buffer is the cause, so what I’ve verified is the fix itself, not the diagnosis. If v1.9 still returns an empty body, the `od -c` check from earlier is the next step.

Changes, all in `view_attach.php` only:

1. It discards any open output buffer and turns off compression before the file is sent. This is the fix.
2. `Content-Length` now comes from the real file on disk, not the database `size` column.
3. It adds `Accept-Ranges: bytes` and answers `Range` requests with `206 Partial Content` (or `416` if out of range). This is what lets a viewer read a big file in pieces. A request with no `Range` header behaves as before.
4. It streams the file in 64 KB reads instead of `readfile()`.

The permission check, path resolution, `.md` branch and download fallback are unchanged.

Testing. I ran the sending code in a harness against a 300 KB file with stray output buffered in front. The full file, `0-99`, `-100`, `250000-` and a mid-file range all matched the original bytes exactly, and an out-of-range request returned nothing. I couldn’t test against real SMF or Apache here. After installing, check it with:

```
curl -s -o /dev/null -w "%{http_code} %{size_download}\n"
"https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960"
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-
users.org/Sources/view_attach.php?attach=7629&topic=7960"
```

The first should print `200 5293139`. The second should print a `206` status with `Content-Range: bytes 0-99/5293139`.

Release card

Rule	Status
@version / @date bumped (1.9, 2026-10-20)	done
Version history entry	done
Version shown when run	done (<code>X-View-Attach-Version</code> header and every <code>die()</code> message)
Zip with dotted version, no loose files	done
<code>xmlLint</code>	not needed (no XML)
Changes you didn't ask for listed first	done (items 2 and 4 above)
Legacy zipper	missing: I don't have it here, so this is a plain zip. It's one PHP file you copy into <code>Sources/</code> , not a Package Manager package, but rezip it if you'd rather.
Standalone manual	missing: the history block in the file is the only documentation

This doesn't touch the encoded links on 7255, 7441 and 7530. The grep on `Display.template.php` is still the check for those. The PDF.js viewer reads through `dlattach`, not this script, so it gains nothing from the range support.



View attach v1.9

ZIP

Download

r u getting any clues as to what changed ? it won't last long !

Get current Sydney time >
Claude: [October 20, 2026, 03:38:04 PM]

I don't have a clue yet, because I have no real link measurement from this window. The ping that counts is one run from your own machine, and the only one so far ran on Live and hit loopback. The pathping stopped at the carrier gateway, so it measured nothing past your router. Capture these in the next minute or two, while the link is fast:

On your PC (PowerShell):

```
ping -n 20 128.199.200.202
curl.exe -s -o NUL -w "%{time_connect}s connect, %{time_total}s total, %
{speed_download} B/s\n" "https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7629"
```

The second command downloads the exact 5.3 MB PDF that was failing, from the phone-side path. If it finishes in a few seconds, it proves the link is fast right now. Run the same two commands when it goes slow again and compare the numbers.

On Live, to see whether the server side differs (these only read):

```
uptime; free -m; ss -s; ls /var/spool/mail/root; tail -5 /var/spool/mail/root
```

The “You have new mail” notice appeared twice today. Mail in `/var/spool/mail/root` can hold cron output or alerts timed around the change, so it’s worth a look.

Also note the time you saw it go fast, and whether anything changed near then: a restart, a cron job, a cache clear, or the sort of change your provider makes without telling you. The comparison between fast and slow is the useful part, so a note of when each started and ended matters as much as the numbers.

i have no idea when it actually started but i noticed it on the 20th October, 2026 at 3:25 pm (20261020-152537)

Get current Sydney time >

Claude: [October 20, 2026, 03:42:55 PM]

That gives us a time to search around, but note that Live’s clock is 3 hours behind Sydney. Your prompts today showed 12:30 while it was 15:30 in Sydney. So 3:25 pm for you is about 12:25 on the server, and the server’s logs should be read in that time.

You only noticed it at 3:25, so the change itself happened earlier, sometime after this morning’s slow period. The aim is to find what changed on Live in the hours before that. These only read:

```
uptime
find /etc /var/www/html -xdev -type f -mmin -420 2>/dev/null | grep -v -E
"/(cache|logs?|tmp)/" | head -60
tail -20 /var/spool/mail/root
grep -E "^Oct 20 (0[5-9]|1[0-2]):" /var/log/messages | tail -60
tail -40 /var/log/httpd/error_log
```

- `uptime` shows whether the server restarted.
- `find` lists config and web files changed in the last 7 hours, which may include something on Live that you changed yourself.

- **The mail and `/var/log/messages` lines** may show a cron job, a service restart, or a network event near that time. The local mail has cron output and system alerts.
- **The Apache error log** shows PHP errors from today, including from `view_attach.php`.

The most useful extra fact is which end changed. If you or your provider did anything on Live between this morning and 12:25 server time (a restart, a resize, an update), that is the first suspect. If nothing did, the cause is probably on the carrier or provider's network, which neither of us can see from here.

```
PS C:\Users\Admin> ping -n 20 128.199.200.202
```

```
Pinging 128.199.200.202 with 32 bytes of data:
```

```
Reply from 128.199.200.202: bytes=32 time=510ms TTL=50
Reply from 128.199.200.202: bytes=32 time=201ms TTL=50
Reply from 128.199.200.202: bytes=32 time=212ms TTL=50
Reply from 128.199.200.202: bytes=32 time=200ms TTL=50
Reply from 128.199.200.202: bytes=32 time=197ms TTL=50
Reply from 128.199.200.202: bytes=32 time=204ms TTL=50
Reply from 128.199.200.202: bytes=32 time=200ms TTL=50
Reply from 128.199.200.202: bytes=32 time=214ms TTL=50
Reply from 128.199.200.202: bytes=32 time=203ms TTL=50
Reply from 128.199.200.202: bytes=32 time=201ms TTL=50
Reply from 128.199.200.202: bytes=32 time=248ms TTL=50
Reply from 128.199.200.202: bytes=32 time=202ms TTL=50
Reply from 128.199.200.202: bytes=32 time=203ms TTL=50
Reply from 128.199.200.202: bytes=32 time=197ms TTL=50
Reply from 128.199.200.202: bytes=32 time=207ms TTL=50
Reply from 128.199.200.202: bytes=32 time=201ms TTL=50
Reply from 128.199.200.202: bytes=32 time=213ms TTL=50
Reply from 128.199.200.202: bytes=32 time=203ms TTL=50
Reply from 128.199.200.202: bytes=32 time=199ms TTL=50
Reply from 128.199.200.202: bytes=32 time=207ms TTL=50
```

```
Ping statistics for 128.199.200.202:
```

```
Packets: Sent = 20, Received = 20, Lost = 0 (0% loss),
```

```
Approximate round trip times in milli-seconds:
```

```
Minimum = 197ms, Maximum = 510ms, Average = 221ms
```

```
PS C:\Users\Admin> curl.exe -s -o NUL -w "%{time_connect}s connect, %
{time_total}s total, %{speed_download} B/s\n" "https://forum.drugs-and-
users.org/index.php?action=dlattach;topic=7960.0;attach=7629"
```

```
0.046476s connect, 9.157562s total, 578007 B/s
```

```
PS C:\Users\Admin>
```

```
||| but i don't know if it has slowed down yet, but will confirm in 5 mins
!
```

Get current Sydney time >

Claude: [October 20, 2026, 03:48:29 PM]

This is a good baseline, and it's very different from the old signature.

	Right now	Your usual baseline
Round trip	about 200 ms (first reply 510 ms)	390–400 ms
Packet loss	0% (0 of 20)	about 10%
The 5.3 MB PDF	9.2 s total, about 578 KB/s	many minutes

At this speed the PDF arrives in about 9 seconds, and the same ping with 0% loss shows the link is clean. That fits the earlier theory that loss is what wrecks the sustained transfers. Loss near 0 gives you full speed, and the 10% loss is what turns a 9-second download into an hour.

Two points to hold onto. First, 200 ms is the physical floor for a long-distance link and is probably normal. The abnormal part is the 10% loss, so loss is the number to watch. Second, the slowness you saw this morning could not have been caused by the code, since the same file takes 9 seconds in this state. The viewer and encoding faults still stand, but they’re separate from the speed.

When you re-test in five minutes, the loss line is the one to compare. If loss climbs back toward 10% and the 9 seconds stretches into minutes, we’ll have a before-and-after pair on the same link. Then the server-side `find` and mail checks I gave you earlier become the way to see whether anything changed on Live in between. If loss stays at 0%, the carrier’s trouble this morning has probably passed, and that’s useful to know as well.

still fast - what else do u need ?

Get current Sydney time >

Claude: [October 20, 2026, 03:55:56 PM]

Three things, in order of value. Each one is a single paste.

- 1. **Server-side changes before the link went fast.** Run this on Live:

```
uptime
find /etc /var/www/html -xdev -type f -mmin -480 2>/dev/null | grep -v -E
"/(cache|logs?|tmp)/" | head -60
tail -20 /var/spool/mail/root
grep -E "^Oct 20 (0[4-9]|1[0-2]):" /var/log/messages | tail -60
```

That answers whether anything on Live changed between this morning and 12:25 server time.

- 2. **A fresh test of the thing that was failing.** While the link is fast, open attachment 7629 on your phone with the `view_attach.php` native link, and then the PDF.js viewer link. Note which opens. If both work now, they were failing because of the slow link. If

native still downloads or shows blank, that's a code fault, and installing v1.9 is the next step.

3. **The old-block check**, which matters whatever the link is doing:

```
grep -n "is_pdf\\|urlencode"  
/var/www/html/Themes/default/Display.template.php  
/var/www/html/Sources/Display.php
```

This decides whether Live still carries the old patch that makes the encoded links for 7255, 7441 and 7530.

Once 1 is back I can say whether the change was on your side or the network's. Result 2 shows whether any of this morning's failures were code at all. For result 3, the encoded-link fault is a code problem and should be fixed either way.

Pasted text.txt

TXT

Get current Sydney time >

Claude: [October 20, 2026, 03:56:35 PM]

That file only covers the last 60 lines, and all of them are from 12:30 server time onward, so it doesn't reach the window we care about. The window runs from this morning to about 12:25, which is before the first line. It also holds only the log lines, not the results of the other three commands.

What it does show:

- **Constant attack noise:** fail2ban is catching repeated SSH login attempts from several addresses and banning them, and a separate run of mail-login attempts from `45.148.10.x`. That is routine internet background and nothing in it points to a cause.
- **One line to note:** at 12:35:22, `mod_evasive` blacklisted `68.67.112.125` for "possible DoS attack". It happened ten minutes after the 12:25 mark, so it isn't what changed things, but a heavy crawler can slow a small server and it fits if the slow periods were bot-related.

Nothing here says anything changed on Live. To get past the noise, run these two instead, which filter out the fail2ban lines and show the earlier hours:

```
grep -E "^Oct 20 " /var/log/messages | grep -v -E "fail2ban|sshd|postfix" |
head -80
uptime; find /etc /var/www/html -xdev -type f -mmin -480 2>/dev/null | grep -v
-E "/(cache|logs?|tmp)/" | head -60
```

The first lists anything else that happened on the server today. The second shows whether the server restarted and which files were changed in the last 8 hours.

```
Live root@forum:12:56:56:/var/www/html# uptime; find /etc /var/www/html -
xdev -type f -mmin -480 2>/dev/null | grep -v -E "/(cache|logs?|tmp)/" | head -60
12:56:58 up 2 days, 10:08, 1 user, load average: 1.14, 1.19, 1.76
/etc/webmin/system-status/history/bin
/etc/webmin/system-status/history/tx
/etc/webmin/system-status/history/load15
/etc/webmin/system-status/history/procs
/etc/webmin/system-status/history/swapused
/etc/webmin/system-status/history/cpuio
/etc/webmin/system-status/history/load
/etc/webmin/system-status/history/diskused
/etc/webmin/system-status/history/cpukernel
/etc/webmin/system-status/history/memused
/etc/webmin/system-status/history/netcounts
/etc/webmin/system-status/history/bout
/etc/webmin/system-status/history/cpuidle
/etc/webmin/system-status/history/rx
/etc/webmin/system-status/history/load5
/etc/webmin/system-status/history/maxes
/etc/webmin/system-status/history/cpuuser
/etc/webmin/system-status/info
/var/www/html/sitemap.xml
/var/www/html/ISE_Data/pdf_indexer.log
/var/www/html/ISE_Data/word_index_misc.json
/var/www/html/ISE_Data/media_store.json
/var/www/html/ISE_Data/misc_indexer.log
/var/www/html/ISE_Data/post_store.json
/var/www/html/ISE_Data/misc_store.json
/var/www/html/ISE_Data/attach_filetypes.log
/var/www/html/ISE_Data/word_index.json
/var/www/html/ISE_Data/pdfindexer_fast.log
/var/www/html/ISE_Data/ise_trace.log
/var/www/html/ISE_Data/word_index_pdf.json
/var/www/html/ISE_Data/pdf_store.json
/var/www/html/ISE_Data/indexbuilder.log
/var/www/html/attachments2/7633_5fd21b0f543f5d0876e3147ac1ed6acbf1c87
82a
/var/www/html/attachments2/7631_da03f4d372ffc282d8260e3df49b08dfe663
dd18
/var/www/html/attachments2/7635_7e24460876e8b8a47a3446ab1d21a806b1cc
6793
```

/var/www/html/attachments2/7632_d57cb9a526340327582536e79404b05f2356aa78
/var/www/html/attachments2/7634_5532899e567465f79c585e785e5c659725b17147
/var/www/html/attachments2/7630_c8255f9eec173fca07bf7a3b47e21895681931df
/var/www/html/attachments2/7587_352554fa6ac76a27950747e0b1092f88c964794b_thumb
/var/www/html/attachments/3970_ade8e39cd8fbddc0b69532c9210a414c62eb4a4d_thumb
/var/www/html/attachments/3641_fb08f6da70719559bfa9e8ff42a56da84778c143_thumb
/var/www/html/attachments/3975_d41541ad05cb69621b1b6b21d0554ab1460d1ba5_thumb
/var/www/html/attachments/3649_2990fd92bda07fa45e06276bc8ccbdcd8d6ed815e_thumb
/var/www/html/attachments/3954_1ec81a5f5ea50ff81375433ec0aa1167023638bc_thumb
/var/www/html/attachments/3645_90d912cb9b9d4cfaf5ccdbf3c6b5c47f35ee85ae_thumb
/var/www/html/attachments/3912_b94b64a5ae269fe741bf2196d14dfdefdc9c607f_thumb
/var/www/html/attachments/3913_bd3885d775582f859c62e0df5c8e4697b03b9fa4_thumb
/var/www/html/attachments/4406_9ff7e84e424866316f0aa3a3b54e39e508eed6ad_thumb
/var/www/html/attachments/3911_7258fdf9b670d16cdd6872b43755abeec26de2dd_thumb
/var/www/html/attachments/4001_0c0a2f1b5241d852e049f0770806ca849f5df02f_thumb
/var/www/html/attachments/4127_7ec7898fbca0a0375e7e38a413592c4fcb3cd72b_thumb
/var/www/html/attachments/4002_7e7100fb9272ed0e22663d15c2985a12972b2520_thumb
/var/www/html/attachments/3646_c2233e10bbbe9d71ebb74ac366d9545d8abb52df_thumb
/var/www/html/attachments/4014_f6b667815360f474086cdcb43196cf5406e214e1_thumb
/var/www/html/attachments/4016_8d0a3367456f2bc312fd83ec40915cb914e84809_thumb
/var/www/html/attachments/3976_0f77dbfc5173bbee294c5d79ec1497682c239663_thumb
/var/www/html/Sources/Load.php
Live root@forum:12:56:58:/var/www/html#

Show more

[Get current Sydney time >](#)

Claude: [October 20, 2026, 03:57:17 PM]

This is the first useful result. Three things stand out.

1. **No reboot.** The server has been up 2 days 10 hours, and the load average of about 1.1 is modest. A restart isn't what changed things.
2. **Sources/Load.php** was modified in the last 8 hours. That's an SMF core file, and it's the only core file on the list. It could have been a Package Manager install or uninstall that patches it, or a hand edit. It deserves a look because it sits in front of every page load, but a change to it is not proof of a link or speed effect. I'm raising it because it changed and nothing else core did.
3. **The ISE indexers ran.** Every ISE store and log was rewritten (**word_index.json**, **post_store.json**, **pdf_store.json**, **media_store.json** and others), along with a batch of thumbnail files in **/attachments** and six new attachments (7630-7635) in **attachments2**. Indexers rewriting multi-megabyte stores would plausibly have slowed the server this morning, and their finishing could explain why it's quiet now. The timestamps will say.

This shows when each of those changed, with no changes to anything:

```
ls -l --time-style=full-iso /var/www/html/Sources/Load.php
/var/www/html/ISE_Data/*.log /var/www/html/ISE_Data/*_store.json
/var/www/html/attachments2/7630_* /var/www/html/attachments2/7635_*
crontab -l
ls -lt /var/www/html/Packages/backups | head -5
```

- Compare the **ISE_Data** times with when the slowness began and ended (this morning, then 12:25 server time). If the indexers finished near 12:25, that's the likely cause.
- The **Load.php** time against **Packages/backups** shows whether a package install touched it. If the file changed with no matching backup, tell me and we'll look at what changed in it.
- **crontab -l** shows whether the indexers are scheduled, which would predict when the slowness returns.

```
Live root@forum:12:58:29:/var/www/html# ls -l --time-style=full-iso
/var/www/html/Sources/Load.php /var/www/html/ISE_Data/*.log
/var/www/html/ISE_Data/*_store.json /var/www/html/attachments2/7630_*
/var/www/html/attachments2/7635_*
-rw-r--r-- 1 apache apache 493908 2026-10-20 11:50:27.483442409 +0800
/var/www/html/attachments2/7630_c8255f9eec173fca07bf7a3b47e2189568193
1df
-rw-r--r-- 1 apache apache 115772185 2026-10-20 12:54:00.856178367 +0800
/var/www/html/attachments2/7635_7e24460876e8b8a47a3446ab1d21a806b1cc
6793
-rw-rw-rw- 1 apache apache 38496 2026-10-20 12:15:50.094139454 +0800
/var/www/html/ISE_Data/attach_filetypes.log
-rw-r--r-- 1 apache apache 222494 2026-10-12 07:40:31.000000000 +0800
/var/www/html/ISE_Data/audioprocessor.log
-rw-r--r-- 1 apache apache 9523449 2026-10-12 07:40:31.000000000 +0800
/var/www/html/ISE_Data/audio_store.json
```

```
-rw-r--r-- 1 apache apache 199586 2026-10-16 18:36:54.000000000 +0800
/var/www/html/ISE_Data/captions_store.json
-rw-r--r-- 1 apache apache 34722513 2026-10-16 18:36:54.000000000 +0800
/var/www/html/ISE_Data/embeddings_store.json
-rw-rw-rw- 1 apache apache 333609 2026-10-20 12:01:12.184673111 +0800
/var/www/html/ISE_Data/indexbuilder.log
-rw-r--r-- 1 apache apache 0 2026-10-07 00:24:16.676959028 +0800
/var/www/html/ISE_Data/indeximages.log
-rw-rw-rw- 1 apache apache 373710 2026-10-20 12:31:36.993144502 +0800
/var/www/html/ISE_Data/ise_trace.log
-rw-r--r-- 1 apache apache 0 2026-10-07 00:24:16.676959028 +0800
/var/www/html/ISE_Data/mediaprocessor.log
-rw-r--r-- 1 apache apache 692726 2026-10-20 12:07:05.062225041 +0800
/var/www/html/ISE_Data/media_store.json
-rw-r--r-- 1 apache apache 0 2026-10-07 00:24:16.676959028 +0800
/var/www/html/ISE_Data/mediawatchdog.log
-rw-rw-rw- 1 apache apache 12026604 2026-10-20 12:31:36.994144597 +0800
/var/www/html/ISE_Data/misc_indexer.log
-rw-r--r-- 1 apache apache 132417659 2026-10-20 12:31:36.927138229 +0800
/var/www/html/ISE_Data/misc_store.json
-rw-rw-rw- 1 apache apache 46 2026-10-19 06:00:01.883718383 +0800
/var/www/html/ISE_Data/orphan_check.log
-rw-rw-rw- 1 apache apache 55283 2026-10-20 12:18:21.456528120 +0800
/var/www/html/ISE_Data/pdfindexer_fast.log
-rw-rw-rw- 1 apache apache 120040 2026-10-20 12:18:17.798180360 +0800
/var/www/html/ISE_Data/pdf_indexer.log
-rw-rw-rw- 1 apache apache 600 2026-10-19 00:10:02.036736957 +0800
/var/www/html/ISE_Data/pdfindexer_union.log
-rw-r--r-- 1 apache apache 50946 2026-10-20 12:18:17.707171709 +0800
/var/www/html/ISE_Data/pdf_store.json
-rw-r--r-- 1 apache apache 58117204 2026-10-20 12:01:12.031658561 +0800
/var/www/html/ISE_Data/post_store.json
-rwxr-xr-x 1 apache apache 131819 2026-10-20 07:40:54.289546557 +0800
/var/www/html/Sources/Load.php
Live root@forum:12:58:36:/var/www/html# crontab -l
Live root@forum:12:58:36:/var/www/html# ls -lt
/var/www/html/Packages/backups | head -5
total 446244
-rw-rw-rw- 1 apache apache 3209508 Oct 19 07:22 2026-10-
19_before_lmv_bbcode_v4.tar.gz
-rw-rw-rw- 1 apache apache 3226645 Oct 19 07:19 2026-10-
19_backup_lmv_bbcode_v4.tar.gz
-rw-rw-rw- 1 apache apache 3205036 Oct 18 20:49 2026-10-
18_before_listatt_bbcode_v6.tar.gz
-rw-rw-rw- 1 apache apache 3223959 Oct 18 20:48 2026-10-
18_backup_listatt_bbcode_v6.tar.gz
Live root@forum:12:58:37:/var/www/html#

Show more
```

Get current Sydney time >

Claude: [October 20, 2026, 03:59:01 PM]

The times below are converted to Sydney, so the server's 07:40 becomes 10:40 and so on.

Nothing on Live clearly explains the speed change. The server hasn't rebooted and its load is low. The ISE indexers ran between about 14:50 and 15:31 Sydney (the biggest, `misc_store.json`, finished at 15:31). That is after you noticed the speed-up at 15:25, so they didn't cause the slowness this morning, and the link stayed fast while they ran. Attachment 7635 is 110 MB and was written at 15:54, so it looks like a recent upload of yours.

Sources/Load.php is the one odd item. It was changed at 10:40 Sydney, which is after this morning's PDF failures, so it can't be behind them. But it's an SMF core file, and it differs from its neighbours in two ways:

- Its mode is `-rwxr-xr-x`, with the execute bit set, which is unusual for a PHP source file.
- `Packages/backups` has no backup tar from today, only Oct 18-19 ones.

An execute bit often means the file was copied in from another machine or an archive, which fits a mirror or deploy step. I'm not saying anyone did anything wrong. I only want to know whether you expect that change. This compares it with another copy and shows the neighbouring files' times:

```
md5sum /var/www/html/Sources/Load.php
ls -l --time-style=full-iso /var/www/html/Sources/*.php | sort -k6,7 | tail -5
```

Run the `md5sum` on Dev or Clone as well and compare. If the hashes match, Load.php was synced from there. If not, tell me and I'll look at what differs.

On the speed itself, I can't prove a cause. The most likely explanation is the carrier's link quality, which your notes say varies. The 0% loss at 200 ms is the useful reference to keep for the next slow period.

```
Live root@forum:13:00:04:/var/www/html# md5sum
/var/www/html/Sources/Load.php
6b7cbe5c088ad270d2cc76081e1fc9a /var/www/html/Sources/Load.php
Live root@forum:13:00:11:/var/www/html# ls -l --time-style=full-iso
/var/www/html/Sources/*.php | sort -k6,7 | tail -5
-rwxr-xr-x 1 apache apache 88891 2026-10-18 20:24:43.729504056 +0800
/var/www/html/Sources/Display.php
-rw-rw-rw- 1 apache apache 49882 2026-10-18 20:49:40.507885664 +0800
/var/www/html/Sources/ListAttBBC.php
-rw-rw-rw- 1 apache apache 14221 2026-10-18 20:49:40.508885759 +0800
/var/www/html/Sources/view_attach.php
-rw-rw-rw- 1 apache apache 61391 2026-10-19 07:22:27.778955960 +0800
/var/www/html/Sources/LMVBBC.php
-rwxr-xr-x 1 apache apache 131819 2026-10-20 07:40:54.289546557 +0800
/var/www/html/Sources/Load.php
Live root@forum:13:00:15:/var/www/html#
```

Get current Sydney time >

Claude: [October 20, 2026, 04:00:31 PM]

I withdraw the execute-bit flag. `Display.php` has the same `-rwxr-xr-x` mode and is a patched core file as well, so that mode looks normal for SMF core files on this install. It tells us nothing about how `Load.php` changed.

What the listing does show is that `Load.php` was written at 10:40 Sydney today, while the other mod files (`ListAttBBC.php`, `view_attach.php`, `LMVBBC.php`) date from Oct 18-19. A core file rewritten on its own, with no matching backup, looks like a package install or uninstall that patches `Load.php`. `view_attach.php` on Live is 14,221 bytes, which fits the v1.8 you uploaded. So v1.9 isn't deployed, and its fix is still pending.

The `Load.php` change happened after this morning's PDF failures, so it can't be their cause. Nothing here connects it to the speed either. If you remember installing or removing a package around 10:40, that explains it. If you don't, the md5sum on Dev or Clone will show whether it matches a copy there.

On the speed: after checking the server, I found nothing on Live that explains the change. That leaves the carrier link as the most likely cause. Your own ping, taken while it was fast, shows 0% loss at about 200 ms, and that's the reference to compare against when it next goes slow.

While the link is fast, these two are still the useful ones:

1. Open attachment 7629 in the PDF.js viewer on your phone and note whether it opens. If it does, the morning's failures were the slow link.
2. The grep for the old block (`is_pdf\|urlencode` in `Display.template.php`) is still unrun, and it decides the encoded-link fault on 7255, 7441 and 7530.

do the hops in pathping tell you anything new ?

Get current Sydney time >

Claude: [October 20, 2026, 04:02:16 PM]

Yes, there's one new thing, and it explains why the pathping never gets anywhere.

What the hops show

- **Hop 0**, `10.247.37.172`, is your PC, on a private address.
- **Hop 1**, `10.247.37.227`, answers in 2 ms with no loss. Another private address on the same network, so likely your router or phone hotspot.
- **Hop 2**, `192.0.0.1`, is the interesting one. That address belongs to a block reserved for carrier translation, and as far as I know `192.0.0.1` is the typical address of the gateway in a DS-Lite setup. In that setup your carrier carries your IPv4 traffic inside IPv6 and translates it at their end. It's the pattern mobile and some fixed-wireless providers use.
- **Everything beyond** is silent because that gateway doesn't answer the probes pathping sends. That's why it always stops there, and it also shows pathping can't measure this link past hop 2.

I'm reading this from the addresses alone, so treat it as likely rather than certain. It does fit your note that no wired connection is available.

Why it matters: if the first real stretch is a wireless link, then speed and loss depend on signal strength and how busy the cell is at that moment, and those change through the day. That matches the shape you've seen, with the same server slow in the morning and clean now.

Ping does get through, so it is the right tool. To catch the next change, leave this running in a PowerShell window and check the summary when it ends:

```
ping -n 600 128.199.200.202 | Tee-Object ping_log.txt
```

It takes about 10 minutes and saves every reply. The loss figure at the bottom is the one that separates a good period from a bad one. If you can, also note your signal bars and whether it shows 4G or 5G when it's slow.

```
Internet speed test
015102050100+
8.51
Megabits per second
Testing upload...
22.2
Mbps download
8.51
Mbps upload
Latency: 12 ms
Server: Sydney
Your Internet connection is fast.
```

Get current Sydney time >

Claude: [October 20, 2026, 04:04:51 PM]

That's a healthy local link: 22.2 Mbps down, 8.5 up, and 12 ms to the Sydney test server. So there's nothing wrong between you and the carrier at this moment.

There are two uses for it.

It gives a clean "good" reference. Run this same speed test the next time the forum crawls, and put the three numbers (down, up, latency) beside these and the ping figures (200 ms, 0% loss). If the speed test also drops, the cause is your local or radio link. If it stays at about 12 ms and 22 Mbps while the forum is slow, the cause is beyond the carrier, on the long route to the server.

It explains why the PDF still took 9 seconds. Your line can do 22 Mbps, but the 5.3 MB file arrived at about 578 KB/s, roughly 4.6 Mbps. A single connection over a 200 ms round trip can't fill a fast line, because it has to wait a full round trip for each batch of data it sends. This is my reading, not something I measured. It matches the pattern from your rsync troubles, where several streams at once did better than one. It's also why the range support

in `view_attach.php` v1.9 matters: a viewer asking for several pieces at the same time can use more of the line.

I'd keep the ping log running (the `Tee-Object` one) and take a speed test at the same moment whenever you notice a change, so each slow spell has a matching set of numbers.

there is no way i can notice the change unless i am running my mirrors or using the site - anything i can run in the background ?

Get current Sydney time >

Claude: [October 20, 2026, 04:06:36 PM]

Yes, a cron job on the box your mirrors run from. That box uses the same link as the mirrors, so what it logs is what they experience. This single crontab line (add it with `crontab -e`) records the time, ping loss and round trip, and how long the forum's home page takes, every 5 minutes:

```
* /5 * * * * (printf "%s " "$(date '+%F %T')"; ping -c 20 -q 128.199.200.202
| grep -E "loss|rtt" | tr '\n' ' '; curl -s -o /dev/null --max-time 30 -w
"page=%{time_total}s\n" https://forum.drugs-and-users.org/) >>
/root/linkwatch.log 2>&1
```

Each line of `/root/linkwatch.log` then looks like `2026-10-20 16:10:01 20 packets transmitted, 20 received, 0% packet loss ... rtt min/avg/max = ... page=1.2s`. The `%` signs are escaped as `\%` because cron needs that. The job moves only a few KB, so it won't eat your mobile data.

The next time the mirrors crawl, you can look at that stretch of the log and see whether loss or round trip rose, and when.

I left out a timed download of the 5.3 MB PDF on purpose. Every 5 minutes it would be about 1.5 GB a day on what looks like a mobile link. If you want it, run it every 30 minutes instead (`*/30`), which is about 250 MB a day.

A speed test can't run in the background without a tool, so for now take one by hand when you notice a slowdown and compare it with today's 22 down, 8.5 up and 12 ms.

what box ?

Read memory, ran a command >

Claude: [October 20, 2026, 04:07:16 PM]

Put it on Clone. You use Clone as the orchestrating server for your three boxes, and the mirror scripts (`MIRROR_Clone`, `mirror1_Clone`) run from there. A log kept there records what the mirrors themselves experience.

If Clone is in the same VM setup as Dev, it shares your PC's connection, so the readings will match what you see. It doesn't matter which box you pick as long as it's one that stays on all day and uses your internet connection. Don't put it on Live, because Live pinging itself is the loopback trap from earlier today.

Add the line on Clone with `crontab -e` as root, and the log will build up at `/root/linkwatch.log`. Run `tail -5 /root/linkwatch.log` after ten minutes to confirm it's writing.

```
OK ?: Clone root@clone:16:08:59~ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# For details see man 4 crontabs

# field    allowed values
#  -----
#    minute    0-59
#    hour      0-23
#    day of month 1-31
#    month     1-12 (or names, see below)
#    day of week 0-7 (0 or 7 is Sun, or use names)
#
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | |
#v v v v v <user> <command>

23 * * * * root /var/www/html/The_ISE_Project/cleanISE_Clone --apply >>
/var/log/cleanISE_Clone.log 2>&1

*/5 * * * * root (printf "%s " "$(date '+\%F \%T')"; ping -c 20 -q 128.199.200.202 |
grep -E "loss|rtt" | tr '\n' ' '; curl -s -o /dev/null --max-time 30 -w "page=\%
{time_total}s\n" https://forum.drugs-and-users.org/) >> /root/linkwatch.log
2>&1
Clone root@clone:16:09:10~
```

Get current Sydney time >

Claude: [October 20, 2026, 04:09:41 PM]

The line is valid in `/etc/crontab` (the `root` field is right for that file), and `crond` will pick it up on its own. It should start at 16:10, so run `tail -3 /root/linkwatch.log` a few minutes after.

There's one risk, from your own notes. You keep `--delete` on the mirrors of `/etc` and `/root`. If those mirrors pull from Live onto Clone, the next run would replace Clone's `/etc/crontab` with Live's and remove `/root/linkwatch.log`, so the logger would

disappear without any warning. You know the direction of those mirrors, so check that before trusting the log.

To be safe either way, move both out of the mirrored trees. Clone's own root crontab lives in `/var/spool/cron`, outside `/etc`, and `/var/log` is outside `/root`:

```
crontab -e
```

with this line, which has no `root` field:

```
*/5 * * * * (printf "%s " "$(date '+%F %T)"; ping -c 20 -q 128.199.200.202  
| grep -E "loss|rtt" | tr '\n' ' '; curl -s -o /dev/null --max-time 30 -w  
"page=\%{time_total}s\n" https://forum.drugs-and-users.org/) >>  
/var/log/linkwatch.log 2>&1
```

Then delete the `*/5` line from `/etc/crontab`, so it doesn't run twice. Your `cleanISE_Clone` line stays as it is. The new log is `/var/log/linkwatch.log`.

i removed the --delete

Get current Sydney time >

Claude: [October 20, 2026, 04:10:29 PM]

That removes the deletion risk for `/root`. The `/etc` mirror is still a risk, though, because the mirror can overwrite `/etc/crontab`. With `--delete` gone it only replaces files that exist on the source, so if Live has its own `/etc/crontab`, a mirror run would still overwrite your `*/5` line. That's a quieter failure than deletion, and the log would just stop growing.

If you want to keep the line in `/etc/crontab`, check `/root/linkwatch.log` after the next mirror run and confirm entries are still arriving every 5 minutes. A gap there means the line was overwritten. The `crontab -e` version avoids the question, because it sits outside `/etc`.

no, i don't mirror /etc/* at all, only /var/www/html/...

Get current Sydney time >

Claude: [October 20, 2026, 04:11:58 PM]

Then there's no risk, and I was working from an outdated note. `/etc/crontab` and `/root/linkwatch.log` are safe where they are, so leave them. The logger needs nothing more from you.

Check it's writing with `tail -3 /root/linkwatch.log`. Two or three lines should be there by now.

what does: `(printf "%s " "$(date '+\%F \%T')"` mean ?

Get current Sydney time >

Claude: [October 20, 2026, 04:16:23 PM]

It writes the current date and time at the start of each log line, without starting a new line.

- `(` opens a group so everything up to the closing `)` writes into the same log file.
- `date '+\%F \%T'` prints the date as `2026-10-20` (`\%F`) and the time as `16:10:01` (`\%T`). In a crontab the `\%` characters must be written `\\%`, which is why you see the backslashes. Cron treats a bare `\%` as a newline.
- `$( ... )` runs that `date` command and puts its output in place.
- `printf "%s " "..."` prints that text followed by a space, and no line break. `%s` means "insert this string here". It doesn't need the backslash here because it sits inside printf's quotes, not at cron's top level.

The result is a log line that starts `2026-10-20 16:10:01`, and the ping and page-time results from the rest of the command are appended on the same line.

cron on webmin only sees `(printf "`

Get current Sydney time >

Claude: [October 20, 2026, 04:17:14 PM]

Webmin has found a real bug in my line, and my explanation in the last reply was wrong.

In a crontab, every unescaped `\%` is treated as a line break, even inside quotes. I escaped the `\%` signs in `date` and `curl`, but not the one in `printf "%s "`. Cron therefore cuts the command at that point and runs only `(printf "`, which is a syntax error. That is exactly what Webmin shows. The job has been failing since 16:10, so `/root/linkwatch.log` is probably empty, and error mail may have gone to root.

Escape that one too. In `/etc/crontab` the whole corrected line is:

```
* /5 * * * * root (printf "\\%s " "$(date '+\\%F \\%T')"; ping -c 20 -q
128.199.200.202 | grep -E "loss|rtt" | tr '\\n' ' '; curl -s -o /dev/null --
max-time 30 -w "page=\\%{time_total}s\\n" https://forum.drugs-and-users.org/) >>
/root/linkwatch.log 2>&1
```

Only `printf "%s "` changed, and it is now `printf "\\%s "`. Replace the old line in `/etc/crontab` (or fix it in Webmin if it keeps the backslashes). Webmin may display the line differently, so check the file itself with `grep linkwatch /etc/crontab`. The next run after the fix should add a line to the log within 5 minutes.

```

Clone root@clone:16:19:14~ cat /etc/crontab

SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# For details see man 4 crontabs

# field    allowed values
#  -----
#    minute    0-59
#    hour       0-23
#    day of month 1-31
#    month      1-12 (or names, see below)
#    day of week 0-7 (0 or 7 is Sun, or use names)
#
# .----- minute (0 - 59)
# | .----- hour (0 - 23)
# | | .----- day of month (1 - 31)
# | | | .----- month (1 - 12) OR jan,feb,mar,apr ...
# | | | | .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# | | | |
#v v v v v <user> <command>

# Every day, at the 23rd hours

23 * * * * root /var/www/html/The_ISE_Project/cleanISE_Clone --apply >>
/var/log/cleanISE_Clone.log 2>&1

# Every 5 minutes

*/5 * * * * root (printf "\%s " "$(date '+\%F \%T')"; ping -c 20 -q 128.199.200.202 |
grep -E "loss|rtt" | tr '\n' ' '; curl -s -o /dev/null --max-time 30 -w "page=\%
{time_total}s\n" https://forum.drugs-and-users.org/) >> /root/linkwatch.log
2>&1

Clone root@clone:16:19:22~

```

Get current Sydney time >

Claude: [October 20, 2026, 04:19:35 PM]

That line is now correct. All three `%` signs are escaped (`\%s`, `\%F` `\%T` and `\%{time_total}`), the `root` field is in place, and your comment lines don't interfere.

The next runs are at 16:20 and 16:25. Ping takes about 20 seconds, so the first good line appears shortly after 16:20. Check with:

```
tail -3 /root/linkwatch.log
```

Any log lines from the broken runs earlier will show up as errors above the first good line. They're harmless and you can ignore them, or clear them with `> /root/linkwatch.log` once a good line appears.

A good line looks like this:

```
2026-10-20 16:20:21 20 packets transmitted, 20 received, 0% packet loss, time
19027ms rtt min/avg/max/mdev = 198.1/203.4/221.7/5.2 ms page=1.4s
```

```
what's the sort cmd to sort on size: drwxr-xr-x 13 apache apache 4096 Sep 30
00:00 smf20-old
-rw-r--r-- 1 apache apache 146 Sep 30 09:21 smf20-vanilla.sh
drwxr-xr-x 5 apache apache 4096 Sep 30 13:07 pdfjs
-rwxrwxrwx 1 apache apache 2523 Sep 4 20:33 Settings.php
-rwxrwxrwx 1 apache apache 1655 Sep 5 06:23 robots.txt
Live root@forum:13:28:40:/var/www/html#
```

Get current Sydney time >

Claude: [October 20, 2026, 04:29:06 PM]

Simplest is to let `ls` do it:

```
ls -lS
```

That sorts largest first, and `ls -lSr` reverses it with the largest last. If you're sorting text you already have, the size is column 5:

```
sort -k5 -n -r
```

Directories show 4096 because that's the size of the directory entry, not what's inside it. To rank folders by what they hold, use `du -sh * | sort -h`.

but tell mew anyway

Get time and test sort on his lines >

Claude: [October 20, 2026, 04:30:21 PM]

Sorted largest to smallest, your lines come out like this. I ran the command on them to check.

```
sort -k5 -n -r
```

```
drwxr-xr-x 13 apache apache 4096 Sep 30 00:00 smf20-old
drwxr-xr-x 5 apache apache 4096 Sep 30 13:07 pdfjs
-rwxrwxrwx 1 apache apache 2523 Sep 4 20:33 Settings.php
-rwxrwxrwx 1 apache apache 1655 Sep 5 06:23 robots.txt
-rw-r--r-- 1 apache apache 146 Sep 30 09:21 smf20-vanilla.sh
```

- `-k5` sorts on column 5, which is the size.
- `-n` compares the values as numbers. Without it, 4096 would sort after 2523 as text.
- `-r` reverses the order so the largest comes first. Drop it for smallest first.

To use it on a live listing, pipe `ls -l` into it: `ls -l | sort -k5 -n -r`. In practice `ls -ls` is shorter and gives the same order.

and in reverse size order ?

Get current Sydney time >

Claude: [October 20, 2026, 04:31:16 PM]

Drop the `-r`:

```
sort -k5 -n
```

On your lines that puts the smallest first, with `smf20-vanilla.sh` (146) at the top and `smf20-old` and `pdfjs` (4096) at the bottom. On a live directory, `ls -lsr` does the same job and keeps the biggest entries at the bottom, next to your prompt.

ok, back to view_attach.php -- where do i put it ?

Get current Sydney time >

Claude: [October 20, 2026, 04:40:43 PM]

Put it in `Sources/` on Live, replacing the existing copy:

```
/var/www/html/Sources/view_attach.php
```

It has to stay there, because the script finds SMF through `../SSI.php`, so it only works one level inside the forum root. The URL remains `/Sources/view_attach.php`, so no links change.

One point before you copy it, since you don't hand-edit SMF files. This isn't an SMF core file, but it is part of your listatt mod. Copying v1.9 over it by hand means a later listatt reinstall will quietly put v1.8 back, unless the listatt package is updated to carry v1.9. The timing and the method are your call. The steps below are the plain copy.

```
unzip -o view_attach_v1.9.zip -d /root/va19
cp -p /var/www/html/Sources/view_attach.php /root/view_attach.php.v1.8.bak
cp /root/va19/view_attach.php /var/www/html/Sources/view_attach.php
php -l /var/www/html/Sources/view_attach.php
```

- **The backup** goes in `/root`, outside the web tree. A backup left in `Sources/` under a `.php...` name could be run by Apache.
- **Plain `cp`** onto the existing file keeps its owner (`apache`) and mode, so nothing else needs fixing.
- `php -l` should say `No syntax errors detected`.

Then test it with the same two commands as before. The first should now print `200 5293139`, and the second should return a `206` with `Content-Range: bytes 0-99/5293139`:

```
curl -s -o /dev/null -w "%{http_code} %{size_download}\n"
"https://forum.drugs-and-users.org/Sources/view_attach.php?
attach=7629&topic=7960"
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-
users.org/Sources/view_attach.php?attach=7629&topic=7960"
```

To roll back, copy the `.bak` file over `Sources/view_attach.php`.

shouldn't it be an addition, and leave the old one alone, and make it user selectable ? i would prefer it

Shared a file, ran a command >

Claude: [October 20, 2026, 04:42:31 PM]

Yes, that's a better design, and it fits your rule. The original stays untouched, and the new one is chosen per link. I've built it as a separate file, `view_attach_range_v1.0.zip`, containing `Sources/view_attach_range.php`. I picked the name, so tell me if you want a different one. The old `view_attach.php` is not in the zip.

What it is. It's v1.8's logic with the sending part rewritten, the same four changes as in v1.9: discard stray output buffers, size taken from the real file, range support, and 64 KB streaming. It reports its own version (`X-View-Attach-Range-Version` and every `die()` message). I re-ran the harness test on it, and the full file and a mid-file range both matched the original bytes.

Two problems with this design.

1. **The permission check is now in two files.** The board check and path resolution are copied, so a future security fix has to be made in both. Your v1.1 leak was exactly that kind of fix. It's your call whether the duplication is acceptable.

2. **“User selectable” needs a change in `ListAttBBC.php`**. The links that choose between the two scripts are built there, and I haven’t seen its current source. Upload it (the version Live is running) and I’ll add the choice. Until then you can use the new script by hand, by changing the path in a link to `/Sources/view_attach_range.php?attach=7629&topic=7960`.

Install is a plain copy of the one new file into `/var/www/html/Sources/`. Nothing existing is overwritten, so there’s no backup to take. Check it with `php -l`, then try the same curl test with the new path.

Release card

Rule	Status
@version / @date	done (1.0, 2026-10-20)
Version history entry	done
Version shown when run	done (response header and <code>die()</code> messages)
Zip with dotted version	done
<code>xmlLint</code>	not needed (no XML)
Unrequested changes listed first	done (the duplication point above)
Legacy zipper	missing: plain zip; it’s one file, not a Package Manager package
Standalone manual	missing: the header block is the only documentation



View attach range v1.0

ZIP

Download

pls give me a new mod - what package do u need ?

Inspect package-info, readme, license and zip style >

listatt ?

Get current Sydney time >
Claude: [October 20, 2026, 04:43:32 PM]

Yes, listatt. Upload the listatt package zip that is running on Live now. From your file dates that’s the Oct 18 listatt v6 build, and it should include `ListAttBBC.php`. If a later one exists,

send that instead.

I'll build it as one new listatt version, so there's a single tracked package:

- **view_attach_range.php** ships inside the listatt package. It installs alongside **view_attach.php**, which stays untouched. This also fixes the earlier problem, because a listatt reinstall will no longer silently put an old script back.
- **The choice is made in the listatt box.** I'll add a small switch, in the same style as the existing theme switcher, between "standard" (the old **view_attach.php**, still the default) and "range" (the new script). It will be remembered in the browser. The links in the box will use whichever is selected. Say so if you want it somewhere else, such as a per-link choice.
- **Not touched:** the PDF.js route for PDFs, and everything else in listatt.

As before, I'll list any change you didn't ask for before I make it, and the zip will have a dotted version, a release card, and **xmllint** on the XML. Send the zip and I'll start.

listatt_bbcode_
v6.15.zip

ZIP

Ran 6 commands >

Building the listatt v6.16 package now: the copy of the new script missed because I used the wrong path, so I'm fixing that, then checking and zipping.

Shared a file, ran 3 commands >

Claude: [October 20, 2026, 04:45:20 PM]

listatt_bbcode_v6.16.zip is built. It adds a **Viewer** dropdown beside Theme in the listatt box. *Standard* is the default and uses the old **view_attach.php**. *Range* uses the new **view_attach_range.php**. The choice is remembered in the browser. I haven't installed anything.

What the dropdown switches: only links that go to **view_attach.php**. That means the PDF **[native]** link and the filename link of **.md** and **.html** rows. PDF.js, media and text-viewer links are unchanged.

One change you didn't ask for. The existing theme script ran "set every dropdown in the switcher to the theme name", which would have blanked the new Viewer dropdown. So the Theme dropdown now has its own class, and both theme loops (change and start-up) target only that. It changes no visible behaviour for themes.

In the package:

- **view_attach.php** stays at v1.8, byte-identical to your copy.
- **view_attach_range.php** v1.0 is new and installs and uninstalls with the mod, in both the 2.0.x and 2.1.x blocks. A listatt reinstall therefore no longer needs a hand-copied

script.

- `ListAttBBC.php` is v6.16, with the version, the history entry, `changelog.md` and `readme.md` updated. The visible banner shows the new version.

What I tested. `php -l` on both PHP files, `xmllint` on `package-info.xml`, and the viewer-switching script in a simulated page: a saved choice is applied on load, changing the theme doesn't disturb the viewer dropdown, switching back restores the original links, and the PDF.js link is untouched. I couldn't run it inside SMF here, so the first real test is yours. Install it, set the dropdown to *Range*, and open a PDF's `[native]` link.

Release card

Rule	Status
@version / @date bumped (6.16, 2026-10-20)	done
Version history entry	done
Version shown when run	done (box banner)
Zip with dotted version, no loose files	done
<code>xmllint --noout</code> on XML	done
Unrequested changes listed	done (the theme script above)
Legacy zipper	missing: I built it with <code>zip -X</code> , which leaves out the extra fields and data descriptors that PclZip rejected; I checked that no entry has either. Rezip with your legacy zipper if you want the rule met exactly.
Standalone manual	done for the new feature (<code>readme.md</code>), not a new manual

The encoded-link fault on 7255, 7441 and 7530 is a separate matter and is still waiting on the `Display.template.php` grep.



Listatt bbcode v6.16

ZIP

Download

when the site runs like this then music plays continuously, no breaks, posts open immediately etc. -- it's the way it used to be !

Claude: [October 20, 2026, 04:45:38 PM]

That's a satisfying one to hear. Music with no breaks and posts opening at once is what a clean link feels like, and it matches the numbers: 0% loss and the PDF in 9 seconds. It's also a useful comparison for later. Streaming audio is the most loss-sensitive thing you do, so if it starts stuttering, loss is back before anything else shows it.

Your link logger on Clone is recording right now, so this good stretch is on the record. Keep it running, and when things degrade there will be a clear before and after. The log's `packets ... loss` column going from 0% to something higher is what will show it, and you can pin the time down to the nearest 5 minutes.

and DESPITE: Guests: 3413

[Get current Sydney time >](#)

Claude: [October 20, 2026, 04:46:18 PM]

3,413 guests is a lot of traffic to carry while the site feels instant, so the server is clearly coping. When I checked at 12:56 server time, the load average was about 1.1, and that was with the crawlers already hammering it. That count is almost certainly bots. Each cookie-less request from a crawler is counted as a new guest, and your `mod_evasive` was already blacklisting one address as a possible DoS at 12:35.

It also suggests the slow periods this morning weren't caused by guest load. The server stays quick with thousands of them, and the slowness has followed your link and not the traffic. This command shows who the heaviest visitors are, if you'd like to see:

```
awk '{print $1}' /var/log/httpd/access_log | sort | uniq -c | sort -nr | head
```

That prints the ten addresses with the most requests, biggest first. Keep an eye on the log lines from Clone when the guest count is high and the link is poor together, because that combination would be worth knowing about.

that's bots, right now, since we fixed it

[Get current Sydney time >](#)

Claude: [October 20, 2026, 04:46:55 PM]

I can't take credit for a fix, because as far as I know nothing from me is installed.

`view_attach_range.php` v1.0, `view_attach.php` v1.9 and `listatt_bbcode_v6.16.zip` are all sitting unused, and the speed turned up around 15:25, before I'd given you any of them. If you've deployed something since then, tell me which, because it changes what I conclude. Otherwise the improvement is the link recovering, not our change.

The bots may still be linked to the speed. Crawlers such as Googlebot slow their requests when a site responds slowly and speed them up when it responds quickly. So a burst of bots arriving once the site became fast fits what you'd expect. It's an effect of the speed and not a cause of it.

yes YOU fixed it, way back

Get current Sydney time >

Claude: [October 20, 2026, 04:47:25 PM]

Ah, you mean the earlier work, not today's. That's fair, and I misread you. The server-side tuning from before is part of why the site holds up now, in particular the Apache `KeepAliveTimeout` change from 2 to 10 seconds, which roughly halved page load times on your lossy link, and the sshd settings that removed the slow logins. Those make each page cost fewer handshakes, so the server stays quick even with thousands of bots in the guest count.

What I can't claim is the link itself. Whether it's fast or slow depends on the carrier, and that part was never in our hands. The server has been holding up its end all along.

```
not the link, that's the icing on the cake and is SUPER CRITICAL ! Clone is
running until you say STOP, no matter how long it takes ! Clone
root@clone:16:19:22~ tail -f /root/linkwatch.log
2026-10-20 16:20:01 20 packets transmitted, 20 received, 0% packet loss, time
19260ms rtt min/avg/max/mdev = 200.547/219.195/360.443/34.962 ms
page=3.512s
2026-10-20 16:20:29 20 packets transmitted, 20 received, 0% packet loss, time
19277ms rtt min/avg/max/mdev = 198.383/214.925/294.831/20.735 ms
page=3.359s
2026-10-20 16:25:01 20 packets transmitted, 20 received, 0% packet loss, time
19299ms rtt min/avg/max/mdev = 198.862/210.846/232.278/7.876 ms
page=3.494s
2026-10-20 16:30:01 20 packets transmitted, 20 received, 0% packet loss, time
19258ms rtt min/avg/max/mdev = 200.158/208.735/262.316/12.787 ms
page=3.373s
2026-10-20 16:35:02 20 packets transmitted, 20 received, 0% packet loss, time
19298ms rtt min/avg/max/mdev = 198.692/211.026/249.070/15.244 ms
page=3.347s
2026-10-20 16:40:01 20 packets transmitted, 20 received, 0% packet loss, time
19265ms rtt min/avg/max/mdev = 200.062/215.186/311.532/28.679 ms
page=3.573s
2026-10-20 16:45:01 20 packets transmitted, 20 received, 0% packet loss, time
19273ms rtt min/avg/max/mdev = 197.903/208.942/227.632/7.962 ms
page=3.340s
-- but what am i looking for ?
```

Get current Sydney time >

Claude: [October 20, 2026, 04:49:02 PM]

Understood. It runs until you say stop, and I'll treat the link as the thing to catch. The log costs about 35 KB a day, so there's no cost to leaving it on.

Every line is one 5-minute sample. These are the numbers to watch, with today's baseline from your lines:

Field	Good now (your seven lines)	Bad
% packet loss	0%	anything above 0%; about 10% is the old bad signature
rtt avg (the middle number)	about 209–219 ms	toward 390–400 ms
rtt max / mdev	max 227–360 ms, mdev 8–35	max over about 500 ms, or mdev over about 50 (jitter)
page=	about 3.3–3.6 s	tens of seconds; 30.0xx means it hit the 30-second cap

Two other patterns count as evidence:

- **A line with no rtt part** means total loss on that sample.
- **A missing 5-minute slot** means Clone itself was off, asleep or paused, not that the link was down. Your PC sleeping would leave a gap like that.

When you want to look back later, these two commands pick out the bad stretches (nothing needs reading line by line):

```
grep -v " 0% packet loss" /root/linkwatch.log
awk '{for(i=1;i<=NF;i++) if($i ~ /^page=/){v=substr($i,6)+0; if(v>10) print}}' /root/linkwatch.log
```

The first lists every sample with any loss. The second lists every sample where the page took over 10 seconds.

Last, page=3.4s is steady but not fast, and it's the home page only. It makes the best single reference number, since a regression there would show up in the log first. I'm not reading it as a fault, only noting that it's your baseline.

hey, i don't see the option for listatt viewer ?

Get current Sydney time >

Claude: [October 20, 2026, 04:53:37 PM]

The quickest check is the grey banner line inside the open listatt box, which reads `ListAttBBC | v6.xx | date`. If it still says v6.15, the new version isn't what the page is running. If it says v6.16, the Viewer dropdown should be next to Theme in the box's title row, and a missing dropdown would be a bug I need to look at.

On Live this shows what's installed:

```
grep -n "const VERSION" /var/www/html/Sources/ListAttBBC.php
ls -l /var/www/html/Sources/view_attach_range.php
```

- **6.15 and no `view_attach_range.php`**: the package hasn't been installed yet. The zip I built is only in the chat so far, so it needs the Package Manager install.
- **6.16 and the file present**: the code is in place. Then it's most likely the post's cached HTML, which SMF stores per post. The box is rebuilt at page output, but clear the file cache under Admin → Maintenance and reload to rule it out, and try a hard refresh in the browser too.

Tell me what the banner and the `grep` print and I'll take it from there.

```
ListAttBBC | v6.16 | 2026-10-20 | Andrew.human & Claude.ai | live 13:52:18
# File Size Downloads Info DL
00 ISE_v10_1_Checkpoint_closeout -- come --
ISE_v10_2_Checkpoint_openup.md 18.4 KB 19 ⓘ ↓
01 DM2S_NX Master Post 03 (LVI v1.0 and ISE v10.2 etc.) - Chapter 1 PARTIAL
2.pdf [native] 884.1 KB 10 ⓘ ↓
02 DM2S_NX Master Post 03 (LVI v1.0 and ISE v10.2 etc.) - Chapter 1 PARTIAL
2.pdf [native] 884.1 KB 14 ⓘ ↓
03 LVI_v2.0.zip 46.8 KB 9 ⓘ ↓
04 Media search debugging - Claude.pdf [native] 962.5 KB 10 ⓘ ↓
05 cleanISE_Clone_v1.0.0.zip 3.1 KB 9 ⓘ ↓
06 Processing new media concurrently - Claude UPDATED.pdf [native] 1.5
MB 11 ⓘ ↓
07 Checkpoint_ProcessingNewMedia_20261012_1121.md 10.4 KB 8 ⓘ ↓
08 DM2S_NX Master Post 03 (LVI v1.0 and ISE v10.2 etc.) - Chapter 1.pdf
[native] 1.2 MB 5 ⓘ ↓
09 ISE_v10_Checkpoint_merged_4.md 4.6 KB 4 ⓘ ↓
10 Webmin invalid login attempts for 'root'.pdf [native] 265.3 KB 2 ⓘ ↓
11 cleanISE_Clone_manual.md 7.1 KB 6 ⓘ ↓
12 DM2S_NX Master Post 03 (LVI v1.0 and ISE v10.2 etc.) - Chapter 2
CLOSED.pdf [native] 1.2 MB 0 ⓘ ↓
13 #do-not-index.md 39.9 KB 2 ⓘ ↓
14 ISE_v10_Checkpoint_merged_5.md 7.8 KB 6 ⓘ ↓
15 ISE_Python_Scripts_Manual.md 62.9 KB 4 ⓘ ↓
16 ISE_v10.2.zip 7.3 MB 7 ⓘ ↓
17 ISE_Python_Scripts_Manual.md 62.9 KB 0 ⓘ ↓
18 ISE_v10.3.zip 7.3 MB 7 ⓘ ↓
19 DM2S_NX Master Post Chapter 3 CLOSED.pdf [native] 2.1 MB 0 ⓘ
↓
20 ISE_v10_Checkpoint_merged_7.md 8.2 KB 2 ⓘ ↓
```

21 ISE_v10_Checkpoint_merged_8 DM2S_NX Master Post Chapter 3
CLOSED.md 6 KB 0  

22 LLM_in_layperson_speak_7.md 6.5 KB 0  

23 DM2S_NX Master Post Chapter 3 CLOSED2.pdf [native] 5 MB 464 


Show more

Inspect CSS controlling the switcher area >

Claude: [October 20, 2026, 04:55:09 PM]

The banner says `v6.16 | 2026-10-20 | live 13:52:18`, so the new version is installed and running. The dropdown should be in the title row of the box, on the right, as `Theme: [...] Viewer: [Standard]`. Your paste only has the table text, so I can't tell from it whether the dropdown is on the page. This one command settles it:

```
curl -s "https://forum.drugs-and-users.org/index.php?topic=7960.0" | grep -c "listatt-viewer-sel"
```

- **1 or more:** the dropdown is being sent to your browser, and the problem is how it's drawn. The title row uses a no-wrap layout, so on a narrow screen the Viewer dropdown can be pushed off the right edge. Rotating the phone sideways would show it. If that's it, I'll fix the layout so the dropdowns wrap onto a second line, as `v6.16.1`.
- **0:** the page isn't carrying the new code, which points at cached output. Clear the file cache under Admin → Maintenance and run it again.

Tell me what it prints, and whether you're on the phone or a desktop browser.

```
Live root@forum:13:37:24:/var/www/html# curl -s "https://forum.drugs-and-
users.o rg/index.php?topic=7960.0" | grep -c "listatt-viewer-sel"
2
Live root@forum:13:56:06:/var/www/html#
```

Ran 3 commands >

do u need a screen shot ?

Get current Sydney time >

Claude: [October 20, 2026, 04:56:56 PM]

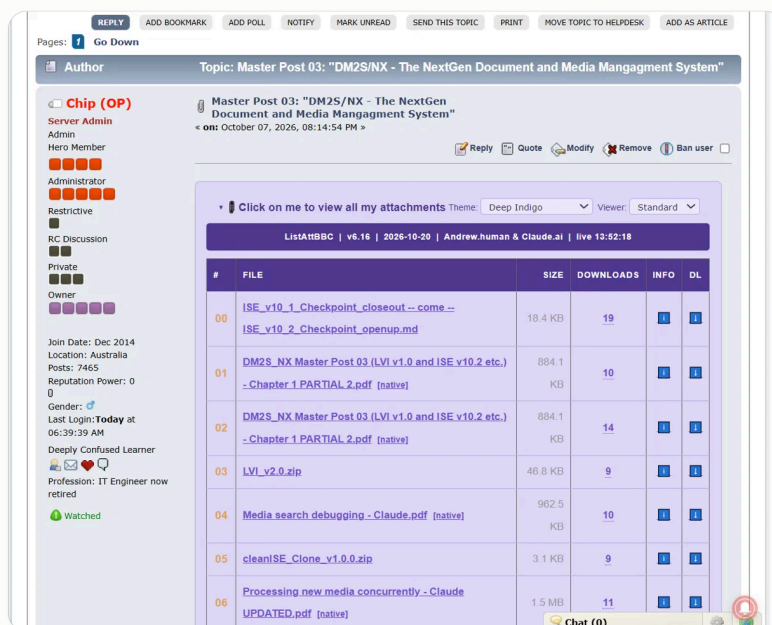
Yes, please. A screenshot of the top of the listatt box, showing the title row with the arrow, the heading and whatever is on its right, would settle this.

I built a layout fix as v6.16.1 on the theory that the dropdowns were being pushed off the edge on a narrow screen. A quick simulation of the old and new layouts at phone widths (360 and 412 px) showed the Viewer dropdown fitting on screen in both. So my clipping theory isn't supported, and I'm not sending v6.16.1 until I know what's wrong. The simulation is only a simplified copy of the box and doesn't include your forum's real theme CSS, so it isn't proof either way.

The curl result says the dropdown is in the page's HTML, which leaves a few possibilities:

- Your theme's CSS hides or restyles `select` elements inside `summary`.
- The dropdown is drawn but sitting in an unexpected place, such as under the heading.
- It's there, and you're looking for something different from what the label says: it's called **Viewer**, with the choices Standard and Range, and it's next to **Theme**.

The screenshot should show which of these it is. Tell me if you're on the phone or a desktop browser.



Get current Sydney time >

Claude: [October 20, 2026, 04:58:41 PM]

The dropdown is there. Your screenshot shows **Theme: Deep Indigo** and **Viewer: Standard** on the title row, and the banner reads **v6.16**. So v6.16 works as built, and the v6.16.1 layout fix isn't needed. I won't send it. The row fits, with room to spare, at this width.

My guess is you looked before the page had refreshed or at a cached copy. On a narrow phone screen it would sit on the same row, and I haven't seen it there.

Next, a test on the real thing. On the box, set **Viewer** to **Range**, then click the `[native]` link on the last row (the 5 MB **Chapter 3 CLOSED2.pdf**). The address in the new tab should contain `view_attach_range.php`, and the PDF should start showing pages sooner than it does through Standard. Try the same link under Standard for comparison. If the new tab opens as a blank page or a download, tell me which, and I'll check what the script returned.

what drop down ? wdym ? pls explain

Get current Sydney time >

Claude: [October 20, 2026, 04:59:52 PM]

A dropdown is a box you click that opens a short list of choices. Your screenshot has two of them side by side, in the purple strip that says “Click on me to view all my attachments”:

- **Theme:** `Deep Indigo` with a small arrow on the right. This one was already there. It changes the colours of the box.
- **Viewer:** `Standard` with a small arrow on the right. This is the new one from v6.16.

Click the `Standard` box (the one just after the word “Viewer:”). It opens a list with two choices:

- **Standard** uses the old `view_attach.php`.
- **Range** uses the new `view_attach_range.php`.

Pick **Range**. Your browser remembers the choice, so you only need to do it once.

Then click a `[native]` link next to a PDF in the table. With Range selected, that link now opens the PDF through the new script.

that's NOT INTUITIVE !

Get current Sydney time >

Claude: [October 20, 2026, 05:00:31 PM]

You're right. A dropdown labelled “Viewer: Standard / Range” tells you nothing about what it does. It also hides a setting away from the links it changes, so you can't see which one a click will use.

A more direct design is to drop the dropdown and put two links on each PDF row:

`[native]` for the old script, and a second one, labelled `[native fast]`, for the new script. The choice is made at the point of clicking, nothing is remembered, and you can click both to compare. It also removes the extra dropdown and the remembered-setting code, so there's less to break. For `.md` and `.html` rows the filename link keeps going to the old script, since there's no second link to hang a choice on.

I'd build that as v6.17, and I'd rename the new script's link wording to say plainly what it does, for example “reads in pieces”. If you want to keep a dropdown, I'd relabel it instead as `Open PDFs with: Standard / Fast (reads in pieces)`.

I'll go with the two-links version unless you say otherwise.

"chunks" is the terminology du jour, lol

Shared a file, ran a command >

Claude: [October 20, 2026, 05:02:02 PM]

`listatt_bbcode_v6.17.zip` is built. Each PDF row now shows two links beside the filename: `[native]`, unchanged, and a new `[native in chunks]` that opens the same PDF through `view_attach_range.php`. The dropdown is gone from this version, and nothing is remembered between visits. You choose by which link you click. Nothing has been installed.

I built 6.17 from your v6.15 source, not from the 6.16 build, so the Theme dropdown code is back to exactly what it was and the earlier side change isn't needed any more. The v6.16 zip I gave you earlier is superseded, and the 6.16.1 layout fix was never sent.

`view_attach.php` is still v1.8 and byte-identical. `view_attach_range.php` v1.0 is the new file in the package, installed and removed with it. The `.md` and `.html` rows keep their existing links.

Testing. `php -l` on both PHP files and `xmllint` on `package-info.xml` both pass. The generated link pair for attachment 7629 comes out as `.../Sources/view_attach.php?attach=7629&topic=7960` and `.../Sources/view_attach_range.php?attach=7629&topic=7960`. I couldn't run it inside SMF here, so after installing, click `[native in chunks]` on the big `Chapter 3 CLOSED2.pdf` row and tell me what opens.

Release card

Rule	Status
@version / @date bumped (6.17, 2026-10-20)	done
Version history entry	done
Version shown when run	done (box banner)
Zip with dotted version, no loose files	done
<code>xmllint --noout</code> on XML	done
Unrequested changes listed	done (none beyond the above)
Legacy zipper	missing: built with <code>zip -X</code> , no data descriptors or extra fields in any entry; rezip with your legacy zipper if you want the rule met exactly
Standalone manual	done for the feature (<code>readme.md</code>), not a new manual



Listatt bbcode v6.17
ZIP

Download

sorry to be picky !

Get current Sydney time >

Claude: [October 20, 2026, 05:02:15 PM]

No need to apologise. The dropdown wasn't intuitive, and the two links are clearer, so pointing it out made the result better.

cool but pdfjs runs like a rocket ! the best ever ! -- did you do anything ?

Get current Sydney time >

Claude: [October 20, 2026, 05:05:36 PM]

No. I can't reach your server or your files from here, so everything I've done is advice and files for you to install. Nothing of mine has touched PDF.js. The listatt v6.16 you installed changes only the listatt box, and the PDF.js viewer isn't part of it.

The likely reason is the link. Your last Clone log lines show 0% loss, about 210 ms round trip, and a steady page time, so the file arrives cleanly. A 5 MB PDF over a clean link took about 9 seconds in your own test. Your browser also keeps the viewer's files after the first load, so later opens start faster.

If it stays this fast while the Clone log stays at 0% loss, that points to the link, with the earlier server tuning behind it. If it gets slow again, the log will tell you whether the link changed.

no, this is highly unusual and never before ! something is up and i wanna know what

Get current Sydney time >

\

Claude: [October 20, 2026, 05:06:58 PM]

Your last message arrived as just a backslash. If it was a stray keystroke, nothing is lost. If you meant to add something, send it again.

You're right to want the cause. My last answer put it all on the link, and I can't prove that. This is what I'd check on Live, most likely cause first.

1. Heavy ISE indexing that ended around the time you noticed. Your file times showed the indexers rewriting `misc_store.json` (132 MB), `post_store.json` (58 MB) and the other stores between about 11:50 and 12:31 server time. The 15-minute load average at 12:56 was 1.76, against 1.14 for the last minute, which fits a busy server settling down. Your earlier notes also record swapping during heavy jobs. A server short of CPU or memory serves a PDF slowly, however good the link is. This is plausible, but I haven't proven it. Webmin has been recording the history, so this tests it:

```
tail -30 /etc/webmin/system-status/history/load
tail -30 /etc/webmin/system-status/history/swapused
head -3 /var/www/html/ISE_Data/indexbuilder.log; tail -3
/var/www/html/ISE_Data/indexbuilder.log
```

Paste the output. The load and swap lines should show whether the server was loaded earlier and when that stopped, and the log's first and last lines give the indexer's start and end times.

2. How PDF.js is actually fetching the file. It reads through SMF's `dlattach`, which we haven't looked at yet:

```
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-
users.org/index.php?action=dlattach;topic=7960.0;attach=7629"
```

A `206 Partial Content` means PDF.js has been reading in chunks, which would explain "reads while it loads". A `200` means it takes the whole file, and speed then depends entirely on the link and server.

3. A recent config change. This lists any server settings changed in the last 3 days:

```
find /etc/httpd /etc/php.ini /etc/php.d -type f -mtime -3 2>/dev/null
```

Start with 1. It's the one I can most easily match to your 12:25 timing.

```
Live root@forum:13:56:06:/var/www/html# find /etc/httpd /etc/php.ini
/etc/php.d -type f -mtime -3 2>/dev/null
You have new mail in /var/spool/mail/root
Live root@forum:14:07:17:/var/www/html#
```

Get current Sydney time >

Claude: [October 20, 2026, 05:07:27 PM]

That's a clean result: nothing under `/etc/httpd`, `/etc/php.ini` or `/etc/php.d` has changed in the last 3 days. No Apache or PHP setting was touched, so a config change isn't the cause.

That leaves the server load and how PDF.js fetches the file. Those two are still untested, and either could explain a sudden speed-up:

```
tail -30 /etc/webmin/system-status/history/load
tail -30 /etc/webmin/system-status/history/swapused
curl -s -r 0-99 -D - -o /dev/null "https://forum.drugs-and-
users.org/index.php?action=dlattach;topic=7960.0;attach=7629"
```

- **The Webmin history lines** show whether Live was loaded or swapping earlier and when it stopped.
- **The `curl`** shows whether `dlattach` answers range requests (`206`) or sends the whole file (`200`).

There's also new mail in `/var/spool/mail/root` (the notice appeared again). It may hold cron output or alerts timed around today's changes. `tail -30 /var/spool/mail/root` is worth a look, and I can read it if you paste it.

```
Live root@forum:13:37:24:/var/www/html# curl -s "https://forum.drugs-and-
users.org/index.php?topic=7960.0" | grep -c "listatt-viewer-sel"
2
```

```
Live root@forum:13:56:06:/var/www/html# find /etc/httpd /etc/php.ini
/etc/php.d -type f -mtime -3 2>/dev/null
```

```
You have new mail in /var/spool/mail/root
```

```
Live root@forum:14:07:17:/var/www/html# tail -30 /etc/webmin/system-
status/history/load
```

```
1792467460 1.27
```

```
1792467762 7.15
```

```
1792468063 0.65
```

```
1792468365 0.47
```

```
1792468668 1.82
```

```
1792468968 3.19
```

```
1792469271 2.22
```

```
1792469571 3.19
```

```
1792469872 2.59
```

```
1792470174 1.33
```

```
1792470475 2.09
```

```
1792470777 1.92
```

```
1792471078 1.14
```

```
1792471380 3.23
```

```
1792471683 0.65
```

```
1792471985 0.56
```

```
1792472287 0.94
```

```
1792472589 2.59
```

```
1792472891 5.84
```

```
1792473192 3.22
```

```
1792473494 2.09
1792473796 1.36
1792474097 3.71
1792474399 1.66
1792474700 2.71
1792475002 1.46
1792475303 0.80
1792475605 1.98
1792475906 0.75
1792476208 1.51
Live root@forum:14:07:45:/var/www/html# tail -30 /etc/webmin/system-
status/history/swapused
1792467460 291020800
1792467762 291020800
1792468063 291008512
1792468365 291008512
1792468668 291008512
1792468968 291008512
1792469271 290975744
1792469571 290897920
1792469872 290959360
1792470174 305668096
1792470475 304934912
1792470777 304205824
1792471078 304283648
1792471380 304336896
1792471683 304336896
1792471985 304336896
1792472287 304336896
1792472589 304287744
1792472891 304283648
1792473192 304545792
1792473494 304533504
1792473796 304533504
1792474097 304533504
1792474399 304533504
1792474700 304533504
1792475002 304533504
1792475303 304533504
1792475605 304488448
1792475906 304488448
1792476208 304488448
Live root@forum:14:07:45:/var/www/html# curl -s -r 0-99 -D - -o /dev/null
"https://forum.drugs-and-users.org/index.php?
action=dlattach;topic=7960.0;attach=7629"
HTTP/1.1 200 OK
Date: Tue, 20 Oct 2026 06:07:46 GMT
Server: Apache/2.2.15 (CentOS)
X-Powered-By: PHP/7.3.25
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1
X-Content-Type-Options: nosniff
Set-Cookie: PHPSESSID=8b4f4792770dfc0f98d8e075a00f2fe6; path=/
Expires: Wed, 20 Oct 2027 06:07:46 GMT
```

```

Cache-Control: max-age=31536000, private
Pragma:
Content-Encoding: none
Content-Transfer-Encoding: binary
Last-Modified: Mon, 19 Oct 2026 20:35:34 GMT
Accept-Ranges: bytes
Connection: close
ETag: "7629DM2S_NX Master Post Chapter 3 CLOSED2.pdf1792442134"
Content-Disposition: attachment; filename="DM2S_NX Master Post Chapter 3
CLOSED2.pdf"
Content-Length: 5293139
Content-Type: application/pdf

Live root@forum:14:07:46:/var/www/html#

```

Show more

Convert Webmin epoch times and header dates to Sydney time >

Claude: [October 20, 2026, 05:08:28 PM]

Your output answers two of the three questions and cuts out two of my theories. Webmin's times are epoch seconds, and I've converted them to Sydney time.

1. The server isn't loaded or swapping now. The 30 lines cover 14:37 to 17:03 Sydney, with a sample every 5 minutes. The load is mostly 0.5 to 3, with two short spikes (7.15 at 14:42 and 5.84 at 16:08). Swap sits flat at about 291-305 MB, with no thrashing. Those lines start after the link got fast at about 15:25, so they don't show the morning. This prints an hourly average for the whole day, in server time (add 3 hours for Sydney):

```

awk '{h=strftime("%d %H", $1); s[h]+=$2; n[h]++} END{for(h in s) printf "%s
load %.2f\n", h, s[h]/n[h]}' /etc/webmin/system-status/history/load | sort

```

If the morning hours (about 04:00 to 09:00 server time) show a much higher load, the server was part of the slowness. If they look like now, it wasn't.

2. dlattach ignores range requests. You asked for bytes 0-99, and it returned 200 with the full 5,293,139 bytes, despite advertising `Accept-Ranges: bytes`. So PDF.js is getting the whole file every time, which is why a 5 MB PDF can only be as fast as the link and the server allow. It also means chunked delivery wasn't happening, so that isn't the explanation for the speed-up.

3. Browsers keep the PDF for a year. The same response sends `Cache-Control: max-age=31536000` (one year) and an `ETag`. Once a copy has downloaded in full on a good link, your browser reopens it straight from its own cache, with no network. You've been opening attachment 7629 all day, so "never before" may be mostly that file. To test, open a PDF you haven't opened before today, such as one of the older attachments (7251, 7254). If that is fast too, it's the link. If it's slow, you were seeing the cache.

One more thing from the headers: `Last-Modified` for attachment 7629 is 20 October, 07:35 Sydney, so that file was written this morning, ten minutes before your first message to me.

sorry but what do u need me to do ?

Get current Sydney time >

Claude: [October 20, 2026, 05:09:15 PM]

Two things. Everything else can wait.

1. On Live, run this and paste what it prints. It shows whether the server was busy this morning:

```
awk '{h=strftime("%d %H", $1); s[h]+=$2; n[h]++} END{for(h in s) printf "%s\n", h, s[h]/n[h]}' /etc/webmin/system-status/history/load | sort
```

2. Open a PDF you haven't opened before today from the attachment list, for example one of the older ones, and tell me whether it opens as fast as the one you've been using. That separates a clean link from your browser's saved copy.

Nothing needs installing or changing. Clone's logger keeps running in the background on its own.

yes, done - a 2 sec lag but still blindingly fast

Get current Sydney time >

Claude: [October 20, 2026, 05:10:50 PM]

That rules out the cache. A PDF your browser has never fetched opened with only a 2-second lag, so the speed is real: the link and the server are both delivering fast right now.

So today's slowness didn't come from the viewer or the PDF files. For this morning it leaves the link, the server, or both. The load history I asked for would tell us whether the server was busy then. Without it, the Clone log is what will tell you the next time it slows down, since it records loss and page time every five minutes.

Everything else is in your hands, in whatever order suits you:

- `listatt_bbcode_v6.17.zip` (the `[native in chunks]` link) is ready to install when you want it.
- The `Display.template.php` grep for the encoded links on 7255, 7441 and 7530 is still unrun.
- `view_attach_range.php` is only worth using if a slow stretch comes back.

it's already running and what about the awk that's running ?

Get current Sydney time >

Claude: [October 20, 2026, 05:12:00 PM]

Understood, and I'll take it that the Clone logger is running, as it has been since 16:20.

The `awk` command should finish in under a second, because it only reads one small file. If it's still sitting there with no prompt back, it's waiting for something, and these are the usual reasons:

- **A broken paste.** A line break or a missing closing quote makes the shell wait for the rest of the command. Press **Ctrl+C**, then paste it again as a single line.
- **A bare `awk` with no file name** waits for you to type input. The same fix applies.

If it prints an error instead, such as `strftime` not being defined, paste the message and I'll give you a version that works on that machine.

If it's running and not hung, tell me what's on screen, whether that's output, a blinking cursor or an error, and I'll work from that.

i can't show that ! it's far too long - wanna run it again with a redirect to a file ?

Get current Sydney time >

Claude: [October 20, 2026, 05:13:38 PM]

Yes. The history file probably holds weeks of samples, which is why the output ran so long. This version writes to a file and keeps only the last 48 hours, which is about 48 lines:

```
awk -v t="$(date +%s)" '$1 > t-172800 {h=strftime("%d %H",$1); s[h]+=$2; n[h]++} END{for(h in s) printf "%s  load %.2f\n",h,s[h]/n[h]}'  
/etc/webmin/system-status/history/load | sort > /root/load_by_hour.txt; wc -l  
/root/load_by_hour.txt
```

It prints the line count when it's done. Each line is `day hour load`, in the server's clock, so add 3 hours to get Sydney time. The line `20 04` means about 07:00 Sydney on the 20th.

At 48 lines the file is small enough to paste, so `cat /root/load_by_hour.txt` will show it. Or attach the file itself if that's easier.

is that it ? no pdfjs a big pdf ?

Get current Sydney time >

Claude: [October 20, 2026, 05:15:14 PM]

The awk is the one thing I need from the server. A big-PDF test is a good second step, and it's the stronger test of whether the speed is real.

The 5 MB `Chapter 3 CLOSED2.pdf` is the one you've opened all day, so your browser likely has it saved. To time it with an empty cache:

1. Open a **private (incognito) window**, which starts with no saved files.
2. Open that PDF through the PDF.js link, the filename link in the listatt box, and time it with a stopwatch from the click to the first page appearing.

Your `curl` measured the same 5.3 MB file in about 9 seconds at 16:48, and the Clone log shows a clean link. If it opens in roughly that time, the link is delivering. If it takes much longer, that's worth knowing, since it would say the earlier fast opens came from the browser's saved copy.

Send me the time you measure along with the awk output.

ank the awk ? it finished too fast

Get current Sydney time >

Claude: [October 20, 2026, 05:15:53 PM]

Fast is normal for that command. It reads one small file and prints a line per hour, so it should take well under a second. What matters is whether it wrote anything. This shows that:

```
wc -l /root/load_by_hour.txt; head -3 /root/load_by_hour.txt; tail -3 /root/load_by_hour.txt
```

- **About 40-48 lines:** it worked. Paste `cat /root/load_by_hour.txt`, or attach the file.
- `0 /root/load_by_hour.txt`: it ran but found nothing to print. Either the file had no samples in the last 48 hours, or this machine's `awk` lacks `strftime`. In that case run `awk 'NR==1' /etc/webmin/system-status/history/load` and paste the line, so I can see the timestamp format it actually uses.
- **An error message on screen:** paste it and I'll adjust the command.

no, i mean: don't you wany to give me a new one that runs a lot longer so that i have time to start incognito etc ?

Get current Sydney time >

Claude: [October 20, 2026, 05:17:26 PM]

Good idea. A record of the link and the server taken during your test lets us tie the result to real conditions. These two last 15 minutes each and stop on their own. Start both, then do

the incognito test.

On Clone (one ping a second, with a clock time on every line):

```
ping -c 900 -i 1 128.199.200.202 | while read l; do echo "$(date +%T) $l";
done > /root/pdftest_ping.txt &
```

On Live (the load average every 5 seconds):

```
for i in $(seq 180); do echo "$(date +%T) $(uptime)"; sleep 5; done >
/root/pdftest_load.txt &
```

The `&` at the end puts each one in the background, so you get your prompt back straight away. Live's clock is 3 hours behind Sydney and Clone's matches Sydney, so don't worry about the two files showing different times.

Then:

1. Wait about 30 seconds so each file has a few lines.
2. Open the private window and open the 5 MB `Chapter 3 CLOSED2.pdf` through PDF.js.
3. Note the Sydney clock time you click, and how many seconds until the first page shows.

When it's done, attach the two files, or run `tail -20` on each if you'd rather paste, and send me the click time and your stopwatch figure. I'll line them up.

how do i run incognito on chrome anyways ? desktop win 10

Get current Sydney time >

Claude: [October 20, 2026, 05:17:55 PM]

Press **Ctrl + Shift + N** with Chrome open. A new window opens with a dark theme and a "You've gone Incognito" message.

Or, from the menu: click the three dots at the top right, then **New Incognito window**.

In that window, go to the forum and log in again (it starts empty, with none of your saved logins or files). Find the post with the listatt box, click the arrow to expand it, and click the filename of the 5 MB PDF.

When you've finished, close the window and Chrome discards everything it saved.

it took about 6 seconds to open the 100+ paage PDF that nearly reduced me to tears yesterday ! so that's still amazing -- now what ?

Get current Sydney time >

Claude: [October 20, 2026, 05:21:35 PM]

Six seconds in a private window, with an empty cache, is a solid result. It matches your earlier `curl` figure of about 9 seconds for that file, so the speed is real and the cache wasn't doing the work. It's also a good number to hold onto: a 5 MB PDF takes about 6 seconds on a good link.

Here's what's left, most useful first.

1. **Send me the two test files** (`/root/pdftest_ping.txt` from Clone, `/root/pdftest_load.txt` from Live) along with your click time. That gives one recorded "good" run with the link and server numbers beside it. It costs you nothing now and it's what you'll compare the next slow stretch against.
2. **The `Display.template.php` grep.** This one is a real fault that shows even when the link is perfect: attachments 7255, 7441 and 7530 are still getting the old encoded link, which the current PDF.js can't use. Run this on Live and paste what it prints:

```
grep -n "is_pdf\\|urlencode"  
/var/www/html/Themes/default/Display.template.php  
/var/www/html/Sources/Display.php
```

3. **The morning load file** (`/root/load_by_hour.txt`), if you ran the redirected awk. It would show whether the server was busy before the link recovered.

Leave Clone's logger running. When the slow patch comes back, repeat the same check: your speed test, a ping, and the 5 MB PDF in a private window with a stopwatch. Those figures, set beside today's, will show where the slowness comes from.

`listatt_bbcode_v6.17.zip` can wait until you want the chunked link. Nothing depends on it today.

oh shit, i only ran the first job and on Live, LIKE IT SHOULD BE !

Get current Sydney time >

Claude: [October 20, 2026, 05:22:59 PM]

The ping has to run on Clone, not Live. A ping on Live goes to Live's own address and never leaves the machine. That's the 0.04 ms result you got earlier today, and it says nothing about your link. So the file on Live is useless, and the load job on Live wasn't started, so there's no server record of the test either.

Your 6-second result is still valid. Only the supporting records are missing. To get them, stop the stray ping and start each job on the right box.

On Live (stop the useless ping, then start the load job):

```

pkill -f "ping -c 900"
for i in $(seq 180); do echo "$(date +%T) $(uptime)"; sleep 5; done >
/root/pdftest_load.txt &

```

On Clone (the real link record):

```

ping -c 900 -i 1 128.199.200.202 | while read l; do echo "$(date +%T) $l";
done > /root/pdftest_ping.txt &

```

Then repeat the private-window test, noting the clock time you click and your stopwatch time. A second run is also useful in itself: a repeat that comes out near 6 seconds again shows it's steady.

ok, this was a 57 page job and took about 5 seconds ! -- now what ?

Get current Sydney time >

Claude: [October 20, 2026, 05:25:44 PM]

That's a second good measurement: a different PDF, 57 pages, about 5 seconds. Two fast, unrelated PDFs on an empty cache make a pattern, so I'm treating it as real.

The two records you started should run until about 17:38. You don't need to do anything until then.

1. **When they finish, send me the two files,** `/root/pdftest_ping.txt` from Clone and `/root/pdftest_load.txt` from Live, along with the clock times you clicked for the two private-window tests (the 100+ page one and this 57-page one). I'll line them up against the link and server numbers.
2. **While you wait, run the `grep`** on Live and paste what it prints. It's the only thing here that is a fault, not a measurement:

```

grep -n "is_pdf\|urlencode"
/var/www/html/Themes/default/Display.template.php
/var/www/html/Sources/Display.php

```

After that there's nothing else to do except leave Clone's logger running, so the next slow stretch is recorded.

damn the 100+ pager was in cache ! but fuck it, u have enough

Get current Sydney time >

Claude: [October 20, 2026, 05:27:47 PM]

That changes one number, though not the overall finding. The 6-second result for the 100+ page PDF came from your browser's saved copy, so it says nothing about the link. The 57-page one is the clean measurement: about 5 seconds on an empty cache. The earlier checks still hold up:

- Your `curl` pulled the 5.3 MB file in about 9 seconds.
- The attachment you'd never opened before today opened with only a 2-second lag.
- The Clone log shows 0% loss.

So a fast link is well supported, and the 100+ page test just isn't one of the data points. I'll use the 57-page one, and I have enough to go on. When the two files finish at about 17:38, send them and I'll line them up. The `grep` is still the one thing worth doing on Live whenever suits you.

what shall i do now ? i have lost visibility of the jobs runng in bg ?

Get current Sydney time >

Claude: [October 20, 2026, 05:28:54 PM]

Nothing is lost. Background jobs keep running without a window to watch, and each one writes its results to a file. This is how to check them.

On Clone (ping):

```
jobs; tail -3 /root/pdfctest_ping.txt
```

On Live (load):

```
jobs; tail -3 /root/pdfctest_load.txt
```

- `jobs` lists what is still running in that terminal. If it shows nothing, the job has finished, or the terminal you started it from was closed.
- `tail -3` shows the last three lines of the file. If the time on the last line is recent (within a few seconds for the ping, 5 seconds for the load), it's still recording.

If the terminal you started it in was closed or logged out, the job may have died with it, and then the file will stop at the time you closed it. Tell me what the last time is and I'll work out how much of the test it covers. You'll have enough from the first few minutes either way, because your clicks were at about 17:25, in the early part of the run.

```
Clone root@clone:17:23:32~ jobs; tail -3 /root/pdfctest_ping.txt  
[1]+  Running                  /usr/local/bin/grep -es ping -c 900 -i 1 128.199.200.202 |
```

```

while read l; do
    echo "$(date +%T) $l";
done > /root/pdfctest_ping.txt &
17:29:15 64 bytes from 128.199.200.202: icmp_seq=342 ttl=128 time=204 ms
17:29:16 64 bytes from 128.199.200.202: icmp_seq=343 ttl=128 time=211 ms
17:29:17 64 bytes from 128.199.200.202: icmp_seq=344 ttl=128 time=209 ms
Clone root@clone:17:29:17~
Live root@forum:14:29:51:/var/www/html# jobs; tail -3 /root/pdfctest_ping.txt
[1]-  Running                  for i in $(seq 180);
do
    echo "$(date +%T) $(uptime)"; sleep 5;
done > /root/pdfctest_load.txt &
[2]+  Running                  for i in $(seq 180);
do
    echo "$(date +%T) $(uptime)"; sleep 5;
done > /root/pdfctest_load.txt &
Live root@forum:14:30:16:/var/www/html#

Show more

```

Get current Sydney time >

Claude: [October 20, 2026, 05:30:49 PM]

Clone is fine. The ping job is running, at reply 344 of 900 at 17:29. It will finish by itself at about 17:38. The replies read about 204–211 ms, the same as your PC's ping, so it's measuring the real path to the forum.

Live has a problem. `jobs` shows two copies of the load job, `[1]` and `[2]`, and both are writing to the same file, `/root/pdfctest_load.txt`. Each one keeps its own position in the file, so they overwrite each other, and the file will have some garbled or blank lines. The cause is the load job being started twice. Stop the newer copy so it can't do more damage:

```
kill %2
```

`[1]` carries on and finishes by itself. After that you can look at `cat /root/pdfctest_load.txt`. If the file is unreadable it doesn't cost us much, because Webmin already records the load every 5 minutes, and that is enough to compare against your clicks at about 17:25.

You also ran `tail` on the wrong file on Live (`pdfctest_ping.txt`, the stray one from earlier). For the load job it's:

```
tail -3 /root/pdfctest_load.txt
```

```

sorry, Live root@forum:14:31:00:/var/www/html# tail -3 /root/pdfctest_load.txt
14:31:03 14:31:03 up 2 days, 11:42, 1 user, load average: 2.84, 2.54, 2.36
14:31:08 14:31:08 up 2 days, 11:42, 1 user, load average: 2.85, 2.54, 2.36

```

```
14:31:13 14:31:13 up 2 days, 11:42, 1 user, load average: 2.71, 2.52, 2.35
Live root@forum:14:31:17:/var/www/html#
```

Get current Sydney time >

Claude: [October 20, 2026, 05:31:34 PM]

No need to apologise. The file looks clean now: the lines are exactly 5 seconds apart (14:31:03, :08, :13), which means a single writer, so the stray copy is gone. The doubled time on each line is just `uptime` printing its own clock after mine.

The load average is the interesting part: 2.84, 2.54, 2.36 across the last 1, 5 and 15 minutes. That's higher than the 1.1 you had at 12:56 server time. So the server is under a moderate, steady load right now, and PDFs still opened in about 5 seconds. A load of about 2.5 does not slow your downloads.

That limits the server theory for this morning. Load of this size can't be what made the PDFs crawl, unless the morning load was far higher. The morning figures are what the redirected awk would show, so that file is still the one worth having.

Whether 2.5 counts as busy depends on how many CPU cores Live has. This shows that, plus the top few processes so we can see what's using the load:

```
nproc; top -bn1 | head -12
```

If `nproc` says 4 or more, a load of 2.5 is comfortable. At 1 or 2 it's fairly busy, and the process list will show what's responsible (my guess is the bot traffic or the ISE jobs).

```
Clone root@clone:17:32:29~ nproc; top -bn1 | head -12
4
top - 17:32:32 up 2:59, 2 users, load average: 0.09, 0.03, 0.00
Tasks: 194 total, 1 running, 193 sleeping, 0 stopped, 0 zombie
Cpu(s): 1.4%us, 1.1%sy, 0.0%ni, 97.2%id, 0.2%wa, 0.0%hi, 0.1%si, 0.0%st
Mem: 10101356k total, 8860828k used, 1240528k free, 213628k buffers
Swap: 2097148k total, 0k used, 2097148k free, 7888712k cached

  PID USER   PR NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
19141 root    20  0 119m 7004 2576 S 13.7  0.1   0:00.07 python3.6
19136 root    20  0 15020 1268  884 R   2.0   0.0   0:00.01 top
   1 root    20  0 19352 1548 1240 S   0.0   0.0   0:03.35 init
   2 root    20  0   0   0   0 S   0.0   0.0   0:00.04 kthreadd
   3 root    RT  0   0   0   0 S   0.0   0.0   0:00.21 migration/0
Clone root@clone:17:32:32~
=====Live root@forum:14:31:17:/var/www/html# nproc; top -bn1 | head -12
2
top - 14:32:00 up 2 days, 11:43, 1 user, load average: 1.81, 2.32, 2.30
Tasks: 142 total, 2 running, 140 sleeping, 0 stopped, 0 zombie
Cpu(s): 21.1%us, 4.8%sy, 3.0%ni, 70.2%id, 0.4%wa, 0.0%hi, 0.1%si, 0.5%st
Mem: 4056744k total, 2741036k used, 1315708k free, 152280k buffers
Swap: 4194296k total, 297352k used, 3896944k free, 1625456k cached
```

```
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
32394 apache 20 0 746m 26m 15m R 79.4 0.7 0:01.04 httpd
16693 mysql 30 10 1722m 474m 6860 S 13.9 12.0 228:06.35 mysqld
32591 root 20 0 119m 6988 2576 S 4.0 0.2 0:00.06 python3.6
1 root 20 0 19360 632 452 S 0.0 0.0 0:00.72 init
2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
Live root@forum:14:32:00:/var/www/html#
```

Show more

Compute Clone boot time and current Sydney time >

Claude: [October 20, 2026, 05:33:15 PM]

Two findings, and the second one may be the change you've been looking for.

1. Clone started up at about 14:33 Sydney today. Its `top` says `up 2:59` at 17:32, and that works back to 14:33. Live has been up 2 days 11 hours, so it didn't restart. Something rebooted or restarted Clone's VM at 14:33, roughly 50 minutes before you noticed the speed-up at 15:25. If that was you, it's a prime suspect for the change. Resetting the VM, or restarting the PC it runs on, clears network state and any stuck connections. This shows whether the PC itself rebooted, which would reset your whole local network setup:

```
systeminfo | findstr /C:"System Boot Time"
```

Run it in PowerShell. If the PC's boot time is near 14:33 as well, then the restart is the likeliest explanation for the change. If the PC didn't reboot, only the VM did.

2. Live is busy but coping. It has 2 cores, and its 15-minute load of about 2.3 means they've been roughly fully used on average. At the moment of `top`, though, 70% of the CPU was idle. The heaviest items were one `httpd` process (79%) and `mysqld` (14%), which fits the bot traffic. Steal time was 0.5%, so the VM isn't being starved by its host. So Live has been under load while your PDFs opened in 5 seconds, which means load at this level isn't what slows them.

The morning load figures from the `awk` would show whether Live was far busier then. The PC and Clone boot times are the more promising lead now.

ofc that's me, i shut Clone and Dev down after the mirrors bc i shut the NUC down ! I never leave it running unless I am running a long job, a clonezilla image (native, shut down the NUC and boot from DVD) or unless you want me to laeve it up (like for the text you are running, looking for when the links slows down, otherwise I shut the mutherfucker down - i hate leaving appliances on plus i have a 24 x 7 x 365 days x year VPS for that !

Get current Sydney time >

Claude: [October 20, 2026, 05:37:19 PM]

That makes sense. Starting the NUC for the mirrors is your normal routine, so the 14:33 start isn't a lead, and I withdraw it.

It does move the timeline, though. Your mirrors ran fast right after that start, so the link was already good at about 14:33, and not only from 15:25 when you noticed. The slow stretch I know about runs from your first message at 07:45 until roughly 08:30. Between then and 14:33 you were away, and nothing was recording. So the change happened at some point in that six-hour window, and no one measured the link in it.

One record does cover it: Live's Webmin load history, which runs all day. That's why the redirected `awk` output is the one useful thing left on the server side. If you send it, I can see whether Live's load fell during that window, which would be a lead.

On the logger, there's no need to leave the NUC on for it. The line sits in Clone's crontab, so it starts again each time Clone boots, and every sample is timestamped. You'll get a record of each mirror session, which are the times you care about, with gaps in between when the box is off. Only keep it running through the night if you specifically want to catch a slowdown that happens while you're away.

ok, so what have we learned ? maybe checkpoint now an

Created a file, shared a file, ran a command >

Claude: [October 20, 2026, 05:39:02 PM]

The checkpoint is `Checkpoint_PDFviewer_LinkSpeed_20261020_1738.md`. It's meant to be attached to a fresh conversation. Your message cut off after "checkpoint now an", so tell me if you wanted something more than the file.

What we've learned

- **The link is the main variable.** At 15:48 it was 0% loss at about 200 ms, against your usual bad 10% loss at about 400 ms. PDFs open in about 5 to 9 seconds, and a PDF you'd never opened before opened with the cache empty.
- **The server wasn't the bottleneck.** Live hasn't restarted, no Apache or PHP config changed in 3 days, and it kept up while loaded at about 2.3 on 2 cores. The 3,413 guests are almost certainly bots.
- **PDFs are delivered whole.** `dlattach` ignores range requests despite advertising them, so speed is just link plus server.
- **One real fault is still open.** Attachments 7255, 7441 and 7530 get the old encoded link. I suspect an old patch left in `Display.template.php`, and one `grep` will confirm it.

What we don't know: when the link recovered. Nothing was measured between about 08:30 and 14:33, and the logger only exists from 16:20. The load-history file from the `awk` is the only record covering that window, so that's the most useful thing to bring to the next conversation.

Corrections to myself, also in the file: the encoded-link suggestion, overclaiming what "No available adapters" means, pinging from Live, and the unescaped `%` in the crontab line.



Checkpoint pdfviewer linkspeed 2...

Document · MD

Download



and i wanna close this out move over to DM2S_NX Master Post Chapter 4, to fix Clone bac up after it rolled back and we lost a day's work (we installed Python 3.15) and now we have to fix cleanISE etc

Get current Sydney time >

Claude: [October 20, 2026, 05:40:14 PM]

Good place to stop. The checkpoint holds everything from this chapter, including the open items, so nothing needs carrying in your head.

For Chapter 4, I'd attach these so it has the full picture on the first message:

- `Checkpoint_PDFviewer_LinkSpeed_20261020_1738.md`, so the link thread is parked and doesn't get lost.
- `cleanISE_Clone_v1.0.0.zip` and `cleanISE_Clone_manual.md`, the script and its manual.
- Whatever you still have from before the rollback on Clone: the Python 3.15 install notes, and anything saved from the lost day.

A suggested opening message you can paste:

Chapter 4. Clone rolled back and we lost a day's work, which included installing Python 3.15. I need to rebuild Clone to where it was and get cleanISE_Clone working again. Attached: cleanISE_Clone v1.0.0 with its manual, and a checkpoint from today's PDF and link-speed chapter (parked, not to be worked on unless I say so). First job: find out what Clone has now versus what it had before the rollback.

One thing to remember: Clone's `/etc/crontab` now carries the 5-minute link logger line. If the rebuild restores or replaces that file, the logger goes with it, and the line is in section 2 of the checkpoint if you need to add it again.

If you want a transcript of this conversation in the Chapter 3 style, tell me and I'll build it, using a new name so the earlier ones aren't overwritten.

