logosobscura · 9mo ago · Edited 9mo ago

They're using HDX (Citrix), it's got a few tells, including the key frame drift when there is some network chop. Know plenty of people involved in the design and build of ICA (that begat HDX), so it's just one of those things you pick up when you've been staring at goats for years. We're seeing a recording of a screen, that is displaying remote content. That seems to be being missed on either side of the push pull over this. I'm generally quite skeptical about this but there are some things that make me think they at least acted it out properly. To the point that focusing on the cursor will absolutely lead you up the garden path- because that's not how cursors render, and when remote, it's very much a 'virtual virtual' cursor.

It's generally how it goes in compartment btw. Rarely are they goi g to give you hands on physical access to a device that stores data like this- you have to remote in, those sessions are logged, and if they'd use a screen recorder (which they wouldn't be able to do in a thin client, but go with it)- they'd have detected that as well. Phone at screen is one of the few ways around it, but it's generally kinda... a tell... when you stand in a SCIF with a phone you shouldn't have pointed at a screen.
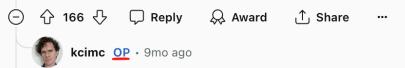
⊖ ⇧ 319 ⇩   💬 Reply   ⌾ Award   ↑ Share   …

Responsible-Local818 · 9mo ago

So the subpixel cursor drift can be caused by a virtual remote cursor? Can you describe how this type of thing works?

⊖ ⇧ 71 ⇩   💬 Reply   ⌾ Award   ↑ Share   …

logosobscura · 9mo ago

Sure, because the protocol has to adapt to network jitter and isn't painted in the remote session, the action is 'click here'. When you're remoting essentially your looking at an image of the screen - let's call it a JOEG for the sake of simplicity- you move your cursor around, and then you interact. We record those movements in the X & Y, and pass on the actions. Here's where the drift can kick in- as the image gets compressed to fuck, and bounced around in real time, artifacts are created (the good protocols hide them well, but they are error correcting)- the drift you see, on a non-local machine, is an absolute tell of EC and the tracking of that to align with the distortions in the rendered frame. When I hear someone talking about frame rates and pixels, in that paradigm, it's kinda missing some really important context.

logosobscura · 9mo ago

Sure, because the protocol has to adapt to network jitter and isn't painted in the remote session, the action is 'click here'. When you're remoting essentially your looking at an image of the screen - let's call it a JOEG for the sake of simplicity- you move your cursor around, and then you interact. We record those movements in the X & Y, and pass on the actions. Here's where the drift can kick in- as the image gets compressed to fuck, and bounced around in real time, artifacts are created (the good protocols hide them well, but they are error correcting)- the drift you see, on a non-local machine, is an absolute tell of EC and the tracking of that to align with the distortions in the rendered frame. When I hear someone talking about frame rates and pixels, in that paradigm, it's kinda missing some really important context.

This could entirely be fake. But it's a good one and one that does get a bit of how things actually work when it comes to the operation of these systems.

⊖ ⌃ 166 ⌄   ▢ Reply   ⧉ Award   ⬆ Share   ···

kcimc OP · 9mo ago

If you can share any screen recordings from the kinds of systems you are referring to, it would help contextualize this a bunch. I looked into the Citrix client and it seems like the default framerate is 30 fps, which already is off from the 24fps in this video.

⊖ ⌃ 75 ⌄   ▢ Reply   ⧉ Award   ⬆ Share   ···

lemtrees · 9mo ago

I'm having a bit of trouble parsing the wording on this documentation from Citrix for XenDesktop 4.0, created in 2014 and updated in 2016.

Near the top they say "With XenDesktop 4 and later, Citrix introduced a new setting that allows you to control the maximum number of frames per second (fps) that the virtual desktop sends to the client. By default, this number is set to 30 fps."

Below that, it says "For XenDesktop 4.0: By default, the registry location and value of 18 in hexadecimal format (Decimal 24 fps) is also configurable to a maximum of 30 fps".

lemtrees · 9mo ago

I'm having a bit of trouble parsing the wording on this documentation from Citrix for XenDesktop 4.0, created in 2014 and updated in 2016.

Near the top they say "With XenDesktop 4 and later, Citrix introduced a new setting that allows you to control the maximum number of frames per second (fps) that the virtual desktop sends to the client. By default, this number is set to 30 fps."

Below that, it says "For XenDesktop 4.0: By default, the registry location and value of 18 in hexadecimal format (Decimal 24 fps) is also configurable to a maximum of 30 fps".

I'm reading the latter to mean that the default FPS for XenDesktop 4.0 was 24 but configurable up to 30, but I'm also reading the former to indicate that the default was 30. This article was made in 2014 and updated in 2016. Maybe it used to be 24, and was later updated to 30 as the default? I've tried looking for older documentation for their legacy software but just get 404s.

⊖ ⬆ 128 ⬇   💬 Reply   🎗 Award   ⬆ Share   ···

> **kcimc** **OP** · 9mo ago · Edited 9mo ago
>
> Holy shit. I think you're reading it right. Citrix was actually running at 24 fps in 2014. If we can find a screen recording from Citrix that shows this subpixel drifting behavior, we may be back out of the fake zone.
>
> Edit: we'd also want to see a zero latency screen recording (the same way that the cursor matches the panned image perfectly without any delay). This implies a server-rendered cursor, and a screen recording that does not include any client cursor. This could be done on the server, or inside the Citrix client app, but not on the client machine itself.

Excellent analysis and I appreciate the effort you put into it. That said, I'm still not convinced that the video is fake. I worked for Maxon for 3 years (the company that makes Cinema 4D), up until last year, although I didn't work on C4D itself (I'm a web developer; I built the Maxon, Cineversity, and C4D Live websites through the agency I worked for), however, I was given a free copy of C4D and full access to everything on Cineversity, so I dabbled quite a bit. I'm not an expert in VFX, but in addition to over 3 years of using C4D as a hobby, I've also had a YouTube channel since 2007 and have a TON of experience with Adobe Premiere and After Effects. As a web developer, I also had extensive professional experience years ago with Flash. So I know my fair share about keyframing and *just* enough about VFX to be dangerous.

The videos were a recording of a screen with a phone. Why would you need to keyframe a mouse cursor instead of just moving your actual mouse cursor around while filming? I get what you're showing and it's somewhat compelling, but at the same time, I feel like there are other plausible explanations for the drifting, and it just doesn't make sense to animate it in post. Why have a mouse moving in there at all, especially if you're going to keyframe it? How would that lend anything to the credibility? How does the mouse move in other parts of the video? Does it move perfectly linear, or is the movement human-like? It would be a pain to make it move human-like with keyframing and seems like a waste of time and effort for something that adds nothing to the video.

Further, if you accidentally delete a keyframe in the middle of two coordinates that are 17px apart, it's not likely to take 214 frames to complete, because 214 doesn't divide evenly into 30 or 60 (two of the most probable frame rates for the original video). Not that it necessarily *has* to be divisible by the frame rate, but it still would have had to have been intentional to have it move that 17px over the course of the 214 frames, whether a keyframe in between was deleted or not.

Other possible explanations: If you have a glass of water or another drink sitting on your desk, assuming everyone nowadays uses optical or laser mice, I've also seen my mouse drift if the condensation from the drink container gets on my desk and under my mouse. Additionally, I've seen it happen when not using a mousepad on a wooden desk. Sometimes the woodgrain, especially if it's even slightly indented, can mess with an optical mouse. I've also just seen it happen with shitty mice, or wireless mice that pick up some minor interference, particularly if they're more than 3 feet away from the receiver, or the battery is low which can weaken the signal. Even aside from all those things, I'm in my 40s and have been using computers almost my entire life, and daily professionally for over 20 years, and I've definitely seen my mouse do weird jumping, drifting, or other strange things without explanation over the years.

Video compression can also have some really weird results that you wouldn't expect. I once filmed a funnel cloud from my backyard, and it was this giant circular cloud that spanned MILES of radius directly in front of me, and the entire thing was rotating at a fairly good speed. In the video though, you could not see the

GrimZeigfeld · 9mo ago

Great post, man. Amazing attention to detail. Just some ideas:

Matching Noise Patterns:

- The identical noise patterns in both videos could potentially be attributed to the sensor noise inherent to a specific satellite. Sensors in space might exhibit consistent noise characteristics under certain conditions. It's possible that if both videos were taken from the same satellite sensor under similar circumstances, they might display matching noise patterns. However, as you mentioned, it is weird that the right video seems to be derived from the left.

Cursor Drift and Text Distortion from Resolution Downsampling:

- The videos may have originally been captured or rendered in ultra-high-resolution (as one might expect from a military reconnaissance satellite). During subsequent viewing or editing back at base, UI elements like cursors, text, and other overlays might have been added.
- If this ultra-high-resolution video with overlays was later downscaled to a lower resolution (exported, screen captured, or filmed), several artifacts could emerge:
    - **Cursor Drift:** The cursor's position in the ultra-high-res version might not perfectly align to the pixel grid in the downscaled version, potentially causing a subpixel drift.
    - **Text Distortion:** Text overlays, if they were anti-aliased or used subpixel rendering, could appear distorted or inconsistent when scaled down.

Frame Interpolation:

- It's also conceivable that if frames were interpolated, perhaps to adjust motion or frame rate for the downscaled version, this could introduce its own set of artifacts, leading to perceived inconsistencies in the footage.

Anyways, excellent post. I'm excited to see what others make of this

⬆ 12 ⬇   💬 Reply   🏅 Award   ⬆ Share   ⋯

⊕ 2 more replies

CharlieStep · 9mo ago · Edited 9mo ago

1. Most cam recorders since the first iphone use some sort of temporal (so multiframe) upsampling and antialiasing techniques that introduce generated noise into the captured image in order to smooth out jaggedness and source data.

2. The video we're analyzing definitely have been preprocessed (i haven't looked into the file codec metadata - but assuming the source was uploaded to YT - we are talking about a vid that has been treated with an custom h264 codec setup) Therefore we should account for some sort of artifacts generated by the codec but also some sort of progressive image stippling / blue noise importance sampling.

3. The artifact ridden doubling of the video feed might be just due to badly written display driver for all we know, or slightly different feed settings (or could be software error). There is a big possibility that artifacts that we're seeing are the result some sort of dithering / smoothing of the feed from the display driver. In such setups that allow for simultanious display of FLIR/color data side to side it could be simply an error or difference in settings in the way that the footage is being feed trough - to the separate image viewports - and there is no way we can check that with the data we have.

Imo - And I hate to be the bummer, because im not ok with the idea of planes being stolen by aliens - the artifacts you portray here can be explained also by the way modern direct data to video solutions work.

Obviously whomever made the videos (assuming they are fake) knew that. Analyzing them on a per pixel basis w/o access to the source data is kinda like looking for signs in a bowl of cereal.

Imo a way more intriguing approach to debunking this video would be to figure out potential lens used to capture the image and then using that to calculate and compare footage with the altitude from which the video feed was allegedly captured. That way i think it'll be fairly easy to see if it checks out with any potential satellite position/data. (unfortunately i dont have time to look into this)

---

traction · 9mo ago

After reading this post, and then the comments about Citrix running at 24fps in 2014 which matches up with this video...WTF.

⊖ ⌃ 7 ⌄   💬 Reply   ⚬ Award   ⬆ Share   ···

> **Glad-Temperature-744** · 9mo ago
>
> I mean, the video was definitely posted in 2014. So Citrix would have been running in 24fps at the time either way.
>
> The question is, who has the knowledge and foresight to even use Citrix to display a video they faked?
>
> There's so many minute aspects here, it would have to be a small team of people who have core knowledge in a bunch of fields directly related to this specific aircraft and the platforms that could have observed it.
>
> ⌃ 3 ⌄   💬 Reply   ⚬ Award   ⬆ Share   ···
>
> ⊕ 2 more replies

Weary-Reading2153 · 9mo ago

OP great write-up, can someone clarify.

   1. The video is either a fake

- Or -

2. It is a contractor remoting into a secure computer over Citrix and using a phone to record what he sees on the screen.

⌃ 6 ⌄   💬 Reply   🎖 Award   ↥ Share   ···

> **Glad-Temperature-744** · 9mo ago
>
> If it's a fake, it appears to be a fake that's being played through a Citrix client somehow. I'd consider it to be nearly impossible to create such a recognizable cursor effect without the actual system. Especially when you could just use the system. I'm just not sure how you'd do that. Run it on a remote server, and then log in, start a pre-rendered video, zoom, and swipe to follow the plane image? That requires you to render a LOT more video. And why? How would they even know that codex is related to this at all?
>
> ⌃ 4 ⌄   💬 Reply   🎖 Award   ↥ Share   ···
>
> ⊕ 2 more replies

mutilatedpuppet · 9mo ago

*someone tries to debunk it with pixel drift

*turns out it proves it is a recording of a physical screen showing a remote session to a government issue operating system further proving the videos as legit.

jesus, this is starting to get scary

every single "debooonk" has only proved them to be more legit

⌃ 6 ⌄   💬 Reply   🎖 Award   ↥ Share   ···

strangelifeouthere · 9mo ago

Bro I think this shit is real LMAOOOOO

⌃ 5 ⌄   💬 Reply   🎖 Award   ↥ Share   ···

AccomplishedWin489 · 9mo ago

Which side are you on? Pro this is a hoax (TIAH) or pro this shit be real(TSBR)? Ive now been on both teams and now back on the Pro TIAH team

⊖ ⌃ 1 ⌄ 💬 Reply ⚬ Award ↑ Share ···

> **kcimc** **OP** · 9mo ago · Edited 9mo ago
>
> First viewing, gut reaction: 90% hoax, 10% real. After considering all the details: 40% hoax, 60% real. After seeing the subpixel cursor: 70% hoax, 30% real. Confirmation of SAR+interferometry techniques, and some app that renders subpixel cursors, would put me back on team real.
>
> ⌃ 11 ⌄ 💬 Reply ⚬ Award ↑ Share ···
>
> ⊕ 11 more replies

Weary-Reading2153 · 9mo ago

>*Could the cursor drift be a glitch from remote desktop software? No strong evidence yet, but here are some suspicions that the remote desktop software Citrix might render a non-OS cursor with subpixel precision and drift glitches. **Remote desktop software doesn't account for the zero latency panning,** but would explain the 24fps framerate.*

Can you explain why the zero latency panning is not possible with the Citrix Client viewer?

- Are you suggesting that it's a slow network connection, and that a remote connection should result in a network latency between the panning mouse and the movement of the map?

If this is a server-side cursor rendering then why would we perceive a latency between the server cursor and the image?

I think the effect is the user panning across and not seeing a response between the physical mouse and the response on the screen. But if it's rendered along with the RDP image there wouldn't be a delay between the cursor and the image. Since they are both server-side rendered.

https://support.citrix.com/article/CTX249907/serverrendered-cursors-performance-analysis-and-optimization

TLDR; The Citrix Client theory accounts for zero latency

⌃ 1 ⌄ 💬 Reply ⚬ Award ↑ Share ···