Managed User Agents – TPAC 2023 Breakout

Minutes for Standardizing managed user agent behavior.

Present

- Reilly Grant (Google Chrome)
- Jeffrey Yasskin (Google Chrome)
- Matt Giuca (Google ChromeOS)
- Simon Hangl (Google ChromeOS)
- Kiara Rose (Apple)
- Erik Anderson (Microsoft Edge)
- Swetha Sivaram (Google ChromeOS)
- Christian Flach (Google ChromeOS, remote)
- David Baron (Google Chrome)
- Alex Christensen (Apple)
- Tim Cappalli (Microsoft Identity)
- Tomislav Jovanović (Mozilla Firefox)

IRC channel

#managed-user-agents

Goal

Discuss the potential for standardization of web-exposed behavior that is controlled by device management features.

Slides

Standardizing managed user agent behavior (TPAC 2023)

Scribes

- Matt Giuca (Google ChromeOS)
- Swetha Sivaram (Google ChromeOS)

Notes

- Thanks to Simon Hangl for finding info on macOS device administration.
- Reilly talks through slides (I won't repeat them here other than adding extra comments).
    - Edge, Firefox and Chrome all support enabling SSO (single-sign-on) through Windows. (Could not find info about Safari.)
    - Jeffrey's privacy principles doc (presented in the session right before this) has a whole section about this. Reilly's words: "The user agent can make decisions on behalf of the device owner, but where it would affect the user, it must disclose that".
    - Permissions state configured by administrator is exposed to the web, and as such observable. Note that this could also be set by the user, but it's slightly different when set by the administrator because no prompt (auto allow or block).
    - Recent APIs proposed that are only available in a managed browser.
- Erik (Microsoft)
    - Generally very supportive.
    - People use native messaging and extensions for such use-cases generally, so there's value in standardizing
    - Careful not to over-specify in some cases. Chrome proposed an API with a very specific collection of fields that the Windows team looked at and they would have chosen a different set of properties. That property bag would be different in each OS and probably should not be part of the standard.
    - Reilly: Please put that feedback on the Device Attributes spec.
    - There are two specs here:
        - Managed Configuration API
        - Device Attributes API – which contains a specific set of properties very specific to ChromeOS and could have been structured in a more flexible way. (This is the one you should provide feedback on.)
- Alex (Apple)
    - Would these APIs be available on all websites always but they just return nothing, other than on managed devices?
    - Reilly: User fingerprinting concern about knowing if you're on a managed device.
    - Alex: In general we are quite hesitant to allow fingerprinting of managed devices. Invites targeting of students and employees of certain organizations. We would prefer not to allow targeting of those vulnerable people.
    - Reilly: An "understandable but naive implementation" would hide it, but maybe not the right thing to do. The Managed Config API (IIRC) exists in all contexts but only provides a non-empty value if the admin has provided a blob. Would that help?
    - Alex: Certainly better than worst case, but as a general thing hesitant to make the web platform be able to tell.
    - Erik (MS): Even if it's the organization wanting to call it out?
    - Alex: That's tough, I'm aware that's a thing but it makes those origins able to fingerprint.

    - Tim Cappalli: Precedent with WebAuthn and Enterprise Attestation with an allowlist of domains.
    - Reilly: This gets into the priority of constituencies. Is this a site the user would be logging into with their enterprise credentials anyway? But still wouldn't expose this in private browsing mode. Being clear to the user that the administrator may be able to see what you do on this particular site because of a configuration in play might be an important privacy principle.
- Simon (Google CrOS)
    - How could other API specs refer to these Privacy Principles principles? And elaborate on them.
    - Jeffrey Yasskin (Google Chrome): If the principle is unclear we should fix it in the Privacy Principles doc.
    - What is the relation between owners and administrators? Where do we draw the line between these 2?
    - Reilly: Lot of possibility for pervasive monitoring which is concerning, but I also understand why administrators need to block or monitor access to specific content. There is a trade-off that we need to make in things that we won't do vs will do but inform the user
- Tim Cappalli (Microsoft identity):
    - Are we thinking about an entire browser, a single profile? What's the context?
    - Reilly: Depends on the implementation. Ties into system mgmt and OS level mgmt. Browsers have benefit, from privacy perspective, of understanding the context that user exists in.
    - Tim: Device level policy and profile level policy tells the user that the browser is managing another profile. See an increase in solutions that need MDM but we should be going in opposite direct in terms of what user actually wants.
    - Reilly: In ChromeOS, significant attempt to make this granular.
    - Tim: Is it worth having 2 definitions - manager or agent?
    - Jeffrey: The privacy principles defines the "user agent" as the profile. Usually profile boundary is important
    - Reilly: How do VPN and proxy settings affect privacy - worth discussing this
- Jeffrey (Google Chrome):
    - Do all enterprise management systems let the admin affect Web APIs.
    - Reilly: Simon and I did some research, I don't believe Safari has any policies that I would consider to affect web APIs in a privacy-concerning way. Firefox does have such policies (e.g. enable permissions, enable SSO).
    - Tomislav Jovanović (Firefox): That's right, though I'm from storage team.
    - Reilly: Chrome is the only one that has its own management infrastructure that isn't just relying on the platform.
    - Jeffrey: If you store your settings in the registry and management can change the registry, then this is all relevant and we need to standardize it. But if everyone else is like Safari then we wouldn't need to standardize since it can't change browser behaviour.
    - Simon: You can disable JS, etc, very coarse grained.

    - Jeffrey: If it can change network traffic and affect the user then that matters.
    - Reilly: Safari supports turning things off like camera permission. You can have MDM configure VPNs. In the network monitoring area outside of the user agent there are setting that allow disabling cross site tracking and other mitigations. Policies exist because legacy enterprise apps need this.
    - Alex: You can say “this domain cannot be accessed from this site”.
    - Jeffrey: We might want to put a carve-out in specs that a managed agent could disable it.
    - Reilly: Discussions about how much should be specified. People often bucket it all into areas where UAs can make decisions, but still helpful to put into the spec.
- Erik:
    - Recently added personal profile - when browser managed at OS level, user can go into policies at the profile level.
    - 2 classes of things to describe in spec - observable and how the managed environment affects the system??
- Matt (ChromeOS): Why would we want to specify?
    - Reilly: HTTP auth and privacy properties around API that make request that may be affected by device configuration and for policy settings that affect how the existing API works - there should be a well-defined behavior when they are managed.
    - Erik: This gives enterprises a way to specify existence of an admin and change what enterprises have permissions to do
    - We could have synchronous way to check with the user, or the enterprise. More of a design principle - not really recommending.
    - Erik: As an example we could make an entry in the federated login spec which said “if you’re in a managed context, then here is where you insert a managed token” or something.
    - Reilly: If we don’t specify some of these things then we could end up seeing behaviour that sites end up relying on which becomes a de facto standard.
    - Enterprises are going to buy products that do this, we should ensure they are build with the ethical web principles. We should ensure we’re involved.
    - Erik: And interoperable.
    - Alex: If one browser vendor has given in to pressure to install trackers doesn’t mean other browser vendors should also cave to that pressure.
    - Reilly: Agree - don’t know history of some of these capabilities. People have tried to bring this up before and then walked away.
    - Alex: We have disappointed some of our corporate enterprise customers in favour of our individual customers. These are business decisions that are taken differently by different businesses.
- Tomislav: I see the most important value of this is making these things explicit, if things are being done against the user’s interest.
    - E.g. informing the user, as a general principle.
    - Jeffrey: Putting it in the spec that “device owners may do this” may put pressure on Apple to do it, but the spec will never *require* device owners to be able to do it.

- Alex:
    - Fingerprinting.
    - Reilly: Important that we get info exposed about managed devices documented as a fingerprinting vector.
    - .: Is it causing unhealthy pressure beyond enterprises?
    - Erik?: Small number of managed device attributes in the browser - doesn't mean this is a small issue
    - We can construct it in specs in a way that clarifies that it is optional, and when it is present, these are the principles upon which it should operate.
    - Alex: I think it can be brought to light in different ways, and not necessarily in W3C specs.
    - Matt: Is your position to not have this concept in spec?
    - Alex: I'm not sure. But concerned about enshrining this and giving it authority. Specs are ultimately supposed to be about interoperability.
- Jeffrey: Some APIs potentially dangerous and users don't have context to evaluate danger. Benefits of being in enterprise environment as an admin, can evaluate danger and when to allow.
    - Matt: Who is danger from whom? If admin protecting a user from malicious site, then admin is in a better position. If admin is attacker, then we shouldn't delegate all power to the administrator.
    - Erik: Note also that admins can make dumb decisions. E.g. don't give admins a global (not per-URL) setting.
    - Reilly: Try to avoid giving footguns to administrators.
    - Erik: Internet Explorer was chock full of those bad policy combinations.
    - Jeffrey (*Chrome engineer speculating on behalf of other browsers, no comment from those browser vendors*): Safari and Firefox haven't shipped WebUSB to everyone. Perhaps they could ship it only behind enterprise policy in order to be usable in companies that use certain hardware.
    - Erik: Enable point-of-sale is another example.
    - Reilly: Some examples on FF where there is a policy to allow installation of extensions.
- Jeffrey: Any conclusions from this?
    - Perhaps: Jeffrey to send a patch to Permissions API to have it address the UA administrator enabling permissions. (cc Tess & maybe Martin)
    - Reilly: Web principles to clarify policies which affect the user agent because they are set by the operating system.
    - Erik: This is going to be controversial, where do we have that discussion?
        - Jeffrey: Was going to test this in the Permissions API patch. Include someone from Safari who are the most skeptical.
        - Tomislav (Mozilla): I am personally supportive, but personal opinion and not speaking for Mozilla. Would like someone else from Firefox to weigh in.
    - Matt: What about the APIs pending that have management as a core part?

- Jeffrey: Might just stick around in the WICG for awhile and see how the discussion goes.