Threat Modeling Report

Created on 11/13/2019 10:34:57 PM

Threat Model Name:

Owner:

Reviewer:

Contributors:

Description:

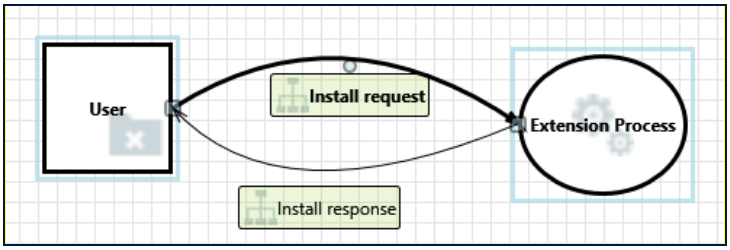
Assumptions:

External Dependencies:

Threat Model Summary:

Not Started	0
Not Applicable	8
Needs Investigation	2
Mitigation Implemented	11
Total	21
Total Migrated	0

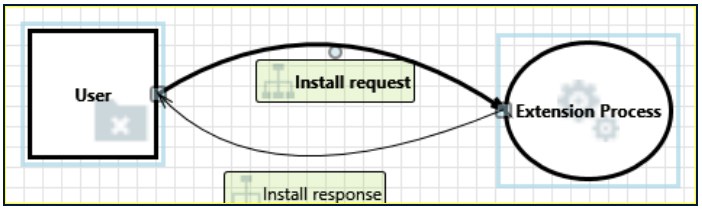
Diagram: ExtensionL0



ExtensionL0 Diagram Summary:

Not Started	0
Not Applicable	0
Needs Investigation	1
Mitigation Implemented	1
Total	2
Total Migrated	0

Interaction: Install request



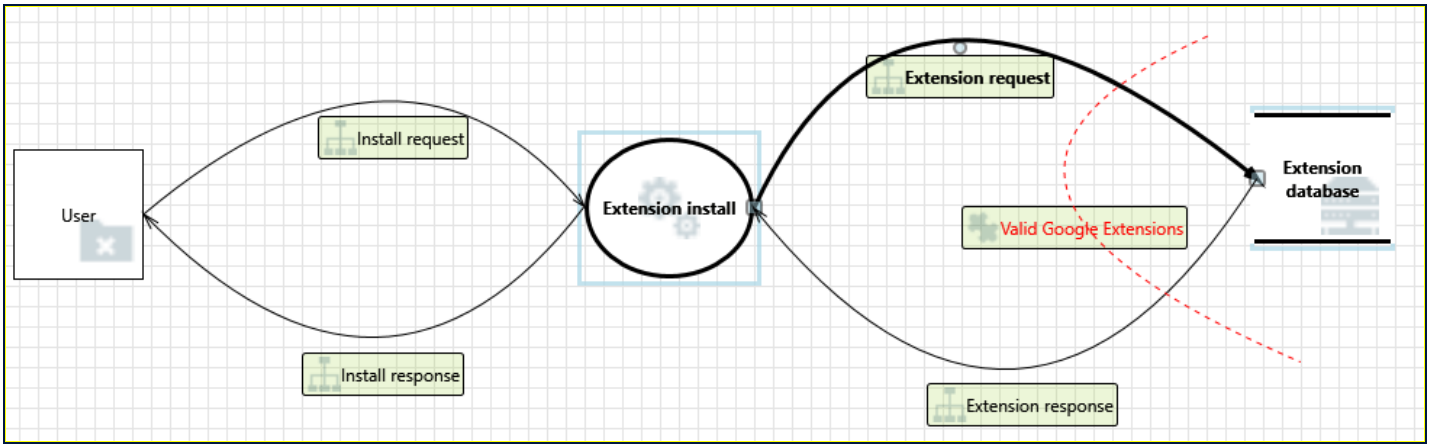
1. Spoofing the User External Entity [State: Mitigation Implemented] [Priority: High]

Category:	Spoofing
Description:	User may be spoofed by an attacker and this may lead to unauthorized access to Extension Process. Consider using a standard authentication mechanism to identify the external entity.
Justification:	User will need to set up and use Chrome Identity API to authenticate. https://developer.chrome.com/apps/app_identity provides uses the instructions to enable with Google and non-Google accounts.

2. Elevation Using Impersonation [State: Needs Investigation] [Priority: High]

Category: Elevation Of Privilege
Description: Extension Process may be able to impersonate the context of User in order to gain additional privilege.
Justification: It seems as though API used for extensions allows for navigation to javascript. This can be used to elevate privlesge for a cross-site scripting attack by a malicious browser extension and can be used to inject content into other extensions. Further investigation will be needed.

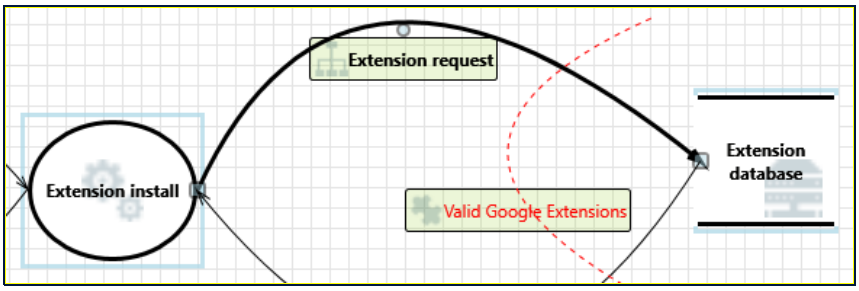
Diagram: ExtensionL1



ExtensionL1 Diagram Summary:

Not Started	0
Not Applicable	8
Needs Investigation	1
Mitigation Implemented	10
Total	19
Total Migrated	0

Interaction: Extension request



3. Spoofing of Destination Data Store Extension database [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: Extension database may be spoofed by an attacker and this may lead to data being written to the attacker's target instead of Extension database. Consider using a standard authentication mechanism to identify the destination data store.
Justification: Since Brave is built on Chromium as a result, it is compatible with all Chromium extensions. (https://support.brave.com/hc/en-us/articles/360017909112-How-can-I-add-extensions-to-Brave-) Nevertheless, Google protects attackers from targeting the extension database through strict policies and guidelines. (https://developer.chrome.com/extensions/single_purpose)

4. Potential Excessive Resource Consumption for Extension install or Extension database [State: Not Applicable] [Priority: High]

Category: Denial Of Service
Description: Does Extension install or Extension database take explicit steps to control resource consumption? Resource consumption attacks can be hard to deal with, and there are times that it makes sense to let the OS do the job. Be careful that your resource requests don't deadlock, and that they do timeout.

Justification: DoS attacks are out of scope in our scenerio. As previously mentioned Brave is built on Chromium, allowing users to use the same extensions. This attack is something Google would take into account. Google has denial of service mechanisms in place to prevent such attacks. (https://www.chromium.org/throttling)

5. Spoofing the Extension install Process [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

Description: Extension install may be spoofed by an attacker and this may lead to unauthorized access to Extension database. Consider using a standard authentication mechanism to identify the source process.

Justification: Extension installs are only acceptable by downloading valid extensions from the Google store. Nevertheless, Google protects attackers from targeting the extension database through strict policies and guidelines. (https://developer.chrome.com/extensions/single_purpose)

6. The Extension database Data Store Could Be Corrupted [State: Mitigation Implemented] [Priority: High]

Category: Tampering

Description: Data flowing across Extension request may be tampered with by an attacker. This may lead to corruption of Extension database. Ensure the integrity of the data flow to the data store.

Justification: As shown in SSL DFD #14, encryption mechanisms in place prevent such attacks. Please refer back to SSL DFD #14 for more information.

7. Data Store Denies Extension database Potentially Writing Data [State: Mitigation Implemented] [Priority: High]

Category: Repudiation

Description: Extension database claims that it did not write data received from an entity on the other side of the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.

Justification: Both Google and Brave have logging in place to help debug if needed. (https://github.com/brave/browser-laptop/blob/master/docs/debugging.md) & (https://support.google.com/chrome/a/answer/6271282?hl=en)

8. Data Flow Sniffing [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure

Description: Data flowing across Extension request may be sniffed by an attacker. Depending on what type of data an attacker can read, it may be used to attack other parts of the system or simply be a disclosure of information leading to compliance violations. Consider encrypting the data flow.

Justification: This attack is out of scope for our Brave browser. However, Google SSL communication lines as shown in our SSL DFD #12. (https://cs.chromium.org/search/?q=chacha&sq=package:chromium&type=cs) & (https://cs.chromium.org/chromium/src/third_party/boringssl/src/ssl/ssl_cipher.cc?q=diffie+boringssl&sq=package:chromium&dr=C)

9. Data Flow Extension request Is Potentially Interrupted [State: Not Applicable] [Priority: High]

Category: Denial Of Service

Description: An external agent interrupts data flowing across a trust boundary in either direction.

Justification: DoS attacks are out of scope for our Brave browser. This is something that Google and Brave systems would handle.

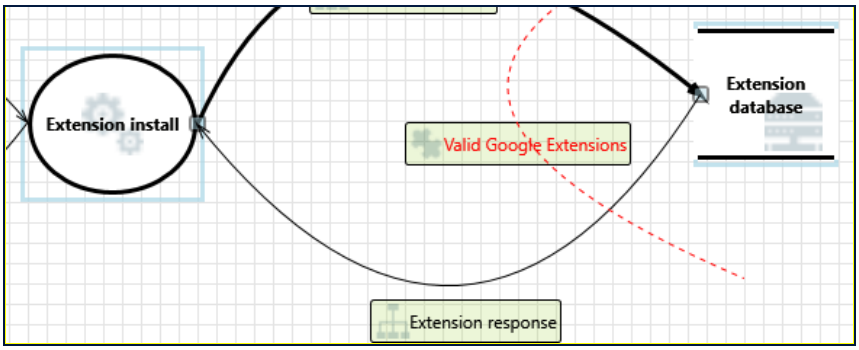
10. Data Store Inaccessible [State: Not Applicable] [Priority: High]

Category: Denial Of Service

Description: An external agent prevents access to a data store on the other side of the trust boundary.

Justification: Once again, DoS attacks are out of scope for our Brave browser. This is something that Google and Brave systems would handle.

Interaction: Extension response



11. Spoofing of Source Data Store Extension database [State: Mitigation Implemented] [Priority: High]

Category: Spoofing
Description: Extension database may be spoofed by an attacker and this may lead to incorrect data delivered to Extension install. Consider using a standard authentication mechanism to identify the source data store.
Justification: Once again, since Brave is built on Chromium, it is compatible with all Chromium extensions. (https://support.brave.com/hc/en-us/articles/360017909112-How-can-I-add-extensions-to-Brave-) Nevertheless, Google protects attackers from targeting the extension database through strict policies and guidelines. (https://developer.chrome.com/extensions/single_purpose)

12. Weak Access Control for a Resource [State: Mitigation Implemented] [Priority: High]

Category: Information Disclosure
Description: Improper data protection of Extension database can allow an attacker to read information not intended for disclosure. Review authorization settings.
Justification: As mentioned previously, Google protects attackers from targeting the extension database through strict policies and guidelines. (https://developer.chrome.com/extensions/single_purpose) However, the extension store is valuable to all users who are looking for extensions to install. This also means any user who follows Googles extension upload process can create and upload an extension. However, their upload process in place helps prevent attacks from uploading malicious content, even though the ability is still there. (https://developer.chrome.com/webstore/publish)

13. Spoofing the Extension install Process [State: Not Applicable] [Priority: High]

Category: Spoofing
Description: Extension install may be spoofed by an attacker and this may lead to information disclosure by Extension database. Consider using a standard authentication mechanism to identify the destination process.
Justification: This attack is out of scope. The extensions are all available for users to see when browsing the Google store; SSL communication lines also help prevent spoofing.

14. Potential Data Repudiation by Extension install [State: Mitigation Implemented] [Priority: High]

Category: Repudiation
Description: Extension install claims that it did not receive data from a source outside the trust boundary. Consider using logging or auditing to record the source, time, and summary of the received data.
Justification: Both Google and Brave have logging in place to help debug if needed. (https://github.com/brave/browser-laptop/blob/master/docs/debugging.md) & (https://support.google.com/chrome/a/answer/6271282?hl=en)

15. Potential Process Crash or Stop for Extension install [State: Not Applicable] [Priority: High]

Category: Denial Of Service
Description: Extension install crashes, halts, stops or runs slowly; in all cases violating an availability metric.
Justification: This attack is also out of scope. DoS attacks are not really handled by the Brave browser. This would be something that Brave systems or Google would handle.

16. Data Flow Extension response Is Potentially Interrupted [State: Not Applicable] [Priority: High]

Category: Denial Of Service
Description: An external agent interrupts data flowing across a trust boundary in either direction.
Justification: Once again, this attack is out of our application, Brave, scope. This attack is something Google systems would handle.

17. Data Store Inaccessible [State: Not Applicable] [Priority: High]

Category: Denial Of Service

Description: An external agent prevents access to a data store on the other side of the trust boundary.

Justification: DoS attacks are out of scope in our scenario. As previously mentioned, Brave is built on Chromium, allowing users to use the same extensions. However, the following site: https://www.chromium.org/throttling talks about how Google mitigates the denial of service attacks.

18. Extension install May be Subject to Elevation of Privilege Using Remote Code Execution [State: Not Applicable] [Priority: High]

Category: Elevation Of Privilege

Description: Extension database may be able to remotely execute code for Extension install.

Justification: This attack is out of scope. This attack is something that would be out of our Brave application scope since the extension database is managed by Google. Nevertheless, Google has mentioned that it protects attackers from targeting the extension database through strict policies and guidelines. (https://developer.chrome.com/extensions/single_purpose)

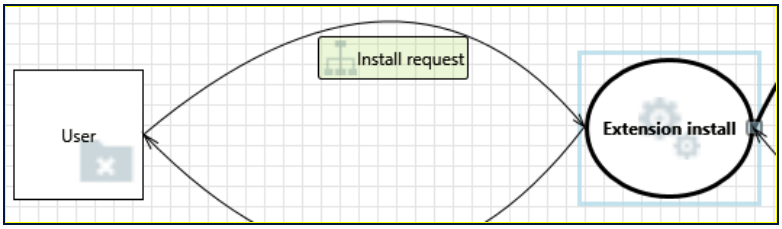
19. Elevation by Changing the Execution Flow in Extension install [State: Mitigation Implemented] [Priority: High]

Category: Elevation Of Privilege

Description: An attacker may pass data into Extension install in order to change the flow of program execution within Extension install to the attacker's choosing.

Justification: Brave mitigates against this attack because of the SSL communication lines it uses. Please see SSL DFD #12 for more information.

Interaction: Install request



20. Spoofing the User External Entity [State: Mitigation Implemented] [Priority: High]

Category: Spoofing

Description: User may be spoofed by an attacker and this may lead to unauthorized access to Extension install. Consider using a standard authentication mechanism to identify the external entity.

Justification: User will need to set up and use Chrome Identity API to authenticate. https://developer.chrome.com/apps/app_identity provides uses the instructions to enable with Google and non-Google accounts.

21. Elevation Using Impersonation [State: Needs Investigation] [Priority: High]

Category: Elevation Of Privilege

Description: Extension install may be able to impersonate the context of User in order to gain additional privilege.

Justification: It seems as though API used for extensions allows for navigation to javascript. This can be used to elevate privlesge for a cross-site scripting attack by a malicious browser extension and can be used to inject content into other extensions. Further review will be needed.