Spyglass Security®
Vulnerability Disclosure

"MetaRisk"

Limited Threat Intelligence Handling Processes Introduce Multiple Threat Vectors to Users of MetaMask Chrome Extension

August 10, 2018

Jackie Stokes | @find_evil
Jason Schorr | @jasonschorr

Executive Summary

Based on an initial assessment using whitelist data[1] and other publicly available information, we believe the MetaMask Google Chrome extension[2]'s current methodology for ingesting and processing threat intelligence does not adequately support the goal of improving user security, and conversely introduces new issues to the ecosystem as evidenced by multiple vulnerabilities identified on July 5-7, 2018 and described in this document. We have titled these issues "MetaRisk" both as a way to refer to them collectively and as a nod to the indirect nature of these vulnerabilities – these are not software bugs, per se, yet still yield an increased attack surface for the project's users.

The majority of these issues hinge on the "Ethereum Phishing Detector"[3] implementation and how this module is intentioned to protect users, especially within context of the practice of whitelisting websites. This practice, in conjunction with a public whitelisting request process, through which the community-at-large can request and easily obtain modifications, can create unsafe conditions for users and other projects which use MetaMask code and the whitelist available via Github and the Infura API[4].

Spyglass Security performed a basic intelligence-focused analysis of three hundred and twenty-two (322) domain names on the whitelist as of July 5, 2018 using Anomali ThreatStream as well as other open and closed sources to gain context and understanding regarding specific hosts and their inclusion on the list. While individuals using this Chrome extension may believe there is an additional layer of personal security to be gained from using MetaMask over other, similar software, hosts included on the whitelist may be gaining additional legitimacy in this manner, thus introducing a new class of vulnerabilities. These style of tactics, designed to gain trust, are seen most prominently within financially motivated cybercrime, whose actors are dependent on phishing as an effective vector to compromise users and secure a foothold on target systems.

While the vulnerabilities described in this disclosure range in severity from informational to high, Spyglass Security as an external third party does not have the necessary context to more fully assess or contextualize threat intelligence processes within MetaMask or ConsenSys. However, we recommend MetaMask immediately develop a remediation plan to address any applicable observations in this report as appropriate.

Vendor Response

We received a near immediate response from the MetaMask project on July 5, 2018 via Twitter[5], and multiple team members, including the lead developer, connected with us personally within two days and responded positively to our outreach. A formal written response was received on July 24, 2018. This document has been updated to include those responses to the findings and our corresponding commentary.

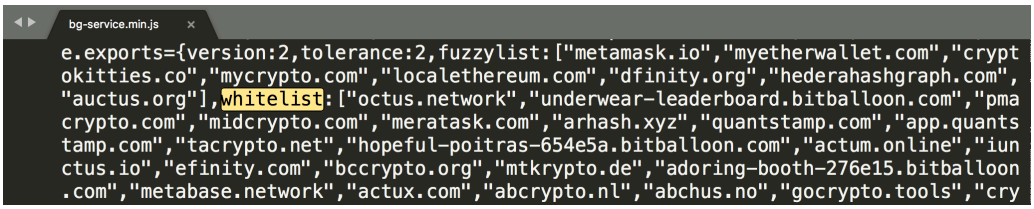
Disclosure Timeline

- July 5 – We requested the right support channel from MetaMask via Twitter[6]
- July 5 – We received a near immediate confirmation of the correct support channel in direct response[7]
- July 7 – Encrypted files sent via email including initial disclosure report and additional data
- July 16 – Update requested
- July 18 – Update received
- July 24 – PGP encrypted response received from MetaMask
- August 8 – Public disclosure draft shared with stakeholders
- August 10 – Public disclosure published

[1] https[:]//github[.]com/MetaMask/eth-phishing-detect/blob/master/src/config.json
[2] https[:]//chrome.google[.]com/webstore/detail/metamask/nkbihfbeogaeaoehlefnkodbefgpgknn?hl=es
[3] https[:]//metamask[.]io/phishing.html
[4] https[:]//api.infura[.]io/v2/blacklist
[5] https[:]//twitter[.]com/metamask_io/status/1014782312758689792
[6] https[:]//twitter[.]com/find_evil/status/1014775516706963461
[7] https[:]//twitter[.]com/metamask_io/status/1014782312758689792

Vulnerabilities

SG-MM-01. Upstream Responsibility Risk

Name	Upstream Responsibility Risk
Identifier	SG-MM-01
Description	Any projects reusing MetaMask's *eth-phishing-detector* code[8] may be putting their users at risk based on their implementation. The MetaMask team should be mindful that other projects will attempt to reuse this code verbatim and adjust accordingly. The *eth-phishing-detector* repository has been forked 108 times[9] as of this writing. • One security-focused Chrome extension reusing *eth-phishing-detector* code has 1,394 users[10]. This particular developer leverages the open Infura API, as shown in Figure 1 below.  *Fig 1. Third party implementation of MetaMask code using Infura API* • Other developers, like Blue Wallet, who may not publicly admit to forking MetaMask code, rely on a static list, as shown in Figure 2 below.  *Fig 2. Third party implementation of MetaMask code using static fuzzy, white, and blacklists, putting users at added risk.*
Recommendation	Be aware implementing a security feature in an open-source project means potentially having some upstream responsibility when other projects reuse code, officially or unofficially. In order to mitigate second-order risk and protect more users, we recommend MetaMask either: A. Add authentication to the Infura API, or B. Only expose hashed values via the API. The code could be reworked to hash each domain visited by the user and compare this hash to the blacklists as with the Google 'Password Alert' Chrome Extension[11]. Doing this would allow the MetaMask team to make the list data private, which will prevent threat actors from easily identifying whitelisted infrastructure.
Vendor Response	

[8] https[:]//github[.]com/MetaMask/eth-phishing-detect/network/members
[9] https[:]//github[.]com/MetaMask/eth-phishing-detect/network/members
[10] https[:]//chrome.google[.]com/webstore/detail/ethersecuritylookup/bhhfhgpgmifehjdghlbbijjaimhmcgnf
[11] https[:]//googleblog.blogspot[.]com/2015/04/protect-your-google-account-with.html

	"It is interesting to consider the upstream responsibility when others reuse our open source phishing list. We will definitely be considering this, and probably improve the list's documentation to help users ensure best usage. From what we can tell, neither recommendation mitigates this problem: • Adding authentication to the Infura API would not add security, because those Infura authentication tokens would be present in our client-side distribution, and easy to reuse for third parties. • While hashing sites could provide privacy benefits from people analyzing our whitelist infrastructure, our whitelist is used to create an automatic blacklist (sites with an unacceptably short levenshtein distance to whitelisted "frequently targeted" sites). This approach has helped us dramatically reduce a large category of phishing sites with familiar looking names, and we are not convinced the list's privacy is so important that we should compromise our fuzzy blacklist. We're also going to consider creating a version of our phishing detector which includes our auto-updating logic, and encourage users to include that library instead of the list module itself."
Our Response	We encourage the wider threat intelligence community to help analyze this issue.

SG-MM-02. Suspicious Whitelisted Infrastructure

Name	Suspicious Whitelisted Infrastructure
Identifier	SG-MM-02
Description	Multiple suspicious hosts were identified in connection with domain names on the whitelist. Two examples are shown here. Additional detail is included in our rough analysts notes. 1. The first domain name we examined, *smbr.rf[.]gd*, is connected to an IP address which was implicated publicly[12] in the EtherDelta hack of December 2017[13]. *rf[.]gd* is a subdomain service provided by *infinityfree[.]net* to users of its free web hosting service. Due to the free nature of the service, it has been abused by malicious actors in the past, likely by either standing up their own malicious website or compromising an unsuspecting subdomain. This specific host has been implicated[14] in threat data collection systems throughout the internet and associated with reams of questionable domain names[15] dating back to 2014. Another domain on the whitelist associated with this host is *fallin.rf[.]gd*.

[12] https[:]//blog.drhack[.]net/etherdelta-hacked-millions-stolen/
[13] https[:]//mashable[.]com/2017/12/21/etherdelta-hacked
[14] http[:]//cybercrime-tracker[.]net/index.php?s=293&m=40&search=Stealer
[15] https[:]//www.virustotal[.]com/en/ip-address/185.27.134.140/information/

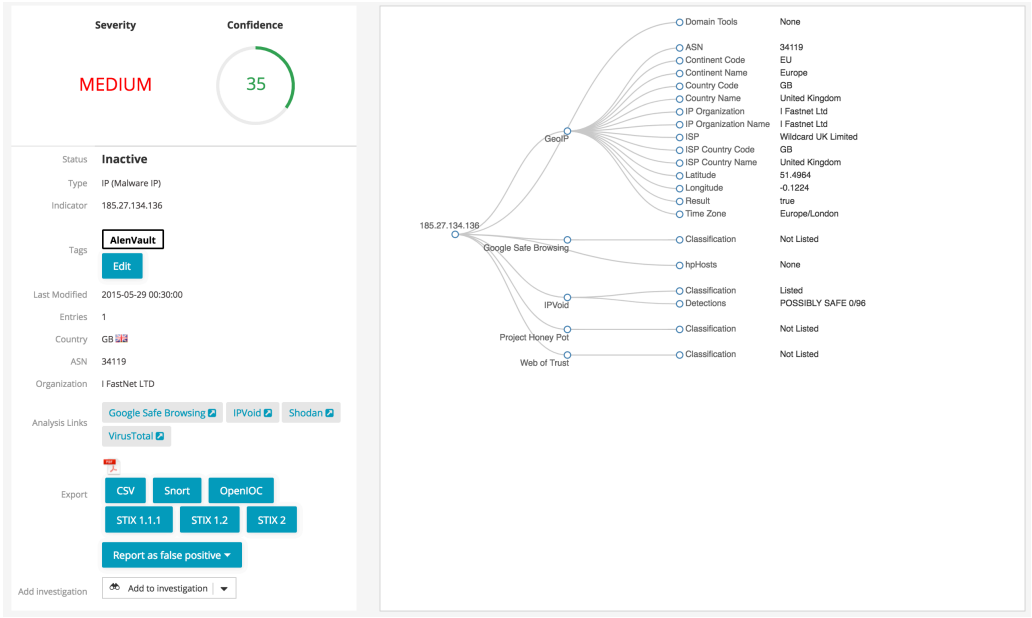


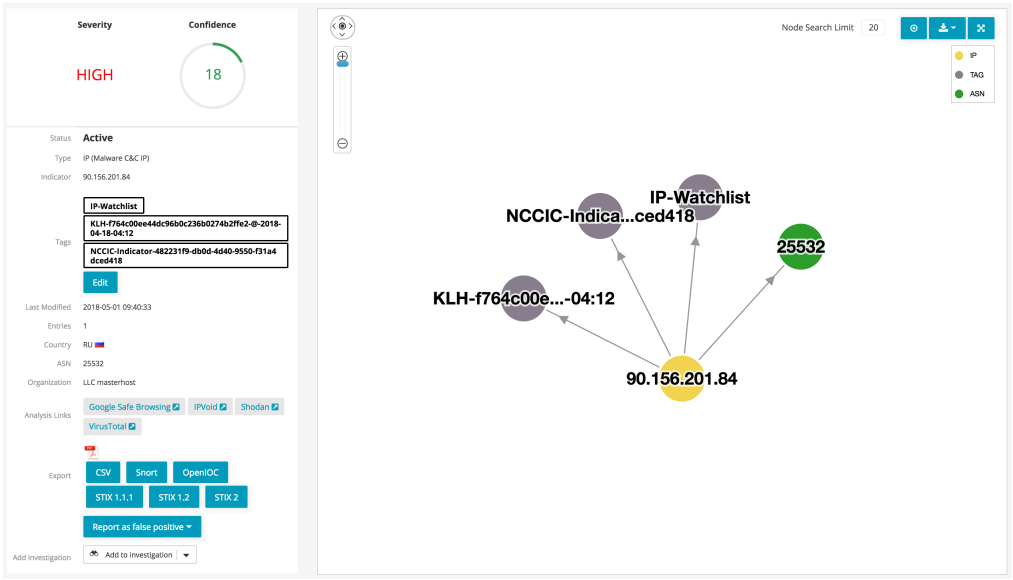
Fig 3. Anomali ThreatStream indicator intelligence summary of 185.27.134.136.

Who are the Hackers of EtherDelta stealing Ethers ?

Currently it is unknown who the hackers are but there are strong possibilities that they may get caught

- Hackers can be traced by looking at the NameServers which were used for DNS Hijacking. Details of Hackers can be accessed after unlocking Hackers used Wildcard Networks Hosting and used **Server IP : 185.27.134.140** which redirected all etherdelta.com visitors to hackers phishing website where they obtained Login Details.
- As visible above, DNS hijacking was not done, rather Cloudflare account of EtherDelta was compromised and A RECORDS were shifted to Hackers website
- Its really strange that it took hours for EtherDelta to respond and all the loss of Coins is totally fault of EtherDelta as they should have kept better login security means.

Fig 4. Screenshot of website showing IP Address publicly implicated in the EtherDelta hack of December 2017.

	2. *metamaks[.]ru*  *Fig 4. Anomali ThreatStream indicator intelligence summary of 90.156.201.84.*
Recommendation	• Remove all domains from the whitelist. • If you cannot stop using the whitelist, remove all domains except those required to enable MetaMask functionality across partner sites.
Vendor Response	"We have done a preliminary investigation, and it appears that the IP addresses you've highlighted as suspicious may have actually been shared infrastructure IP addresses, so we have no current reason to believe they are actually suspect. Additionally, the whitelist is mostly used for blocking additional (similar) domains, and so if these domains were ever reported as malicious, we would expect to block them as soon as possible, like we would any new phishing site. Currently the whitelist almost exclusively exists to allow domains whose names fall within the levenshtein distance of "fuzzylisted" domains. Removing all domains from whitelist does not make sense for us, because it is used to automatically block additional sites. Sites on the whitelist do not have any additional privileges. They are merely non-blocked sites for whom we also automatically block similar domains. We will review the whitelist and prune it to the minimum we can, but some sites need to be whitelisted for different reasons, especially since we are blocking with a levenshtein distance, and so we do not see a blanket removal of sites from the whitelist as a practical step forward."
Our Response	Shared infrastructure of the type mentioned here is weak and often used for years by both skilled and unskilled threat actors to conduct attacks. Several of the systems we specifically identified were running vulnerable versions of publicly-accessible management services, such as cPanel. The examples cited were used to describe the inherent weaknesses underpinning shared infrastructure. While we fundamentally disagree with the way whitelisting is used in this scenario, we encourage the wider threat intelligence community to help analyze this issue. The vendor response to SG-MM-03 below states they will be minimizing use of the whitelist moving forward.

SG-MM-03. Manipulable Whitelisting Process

Name	Manipulable Whitelisting Process
Identifier	SG-MM-03
Description	• Individuals of uncertain origin or legitimacy can request a domain to be whitelisted[16] [17]. • It appears the internal team methodology for assessing a domain as malicious or safe is the exclusive use of *urlscan[.]info* as evidenced by prior commits[18]. This is an inadequate source of threat intelligence information. We assume the reason for implementing a whitelisting process is to mitigate issues created through use of the Levenshtein algorithm to identify potential phishing sites[19], which results in false positives which must be resolved manually once reported as an issue on Github. Improperly classifying a website could have other significant negative downstream repercussions for MetaMask users besides simple availability, as threat actors may seek to compromise whitelisted infrastructure to deliver targeted malware to MetaMask users. Attackers can host innocuous content while undergoing a whitelisting review and modify the site contents at a later date.
Recommendation	• Only highly reputable websites operated by known, established business actors with no history of recent compromise and demonstrated commitment to cybersecurity should be whitelisted. • Implement an industry standard process for deeper security- and intelligence-assessment of each requested domain, or outsource this function. • The whitelist is a critical asset. As such, we recommend implementing raising visibility for the core team when commits are made to the whitelist on Github through alerting or some other process. • Consider completely abandoning the whitelisting process by using a purpose-built system to process threat intelligence, including community submissions, which should be processed in a reliable, systematic, methodological manner consistent with industry best practices. • We recommend evaluating Cisco/OpenDNS's PhishTank API[20] [21] [22].
Vendor Response	"You mention that 'threat actors may seek to compromise whitelisted infrastructure to deliver targeted malware to MetaMask users'. We would be interested to hear about exactly what you mean. Whitelisted sites are not more available than other non-blacklisted sites. We do not list nor recommend them anywhere. Ultimately, our phishing detector heavily relies on fast responses to new threats, and no misconfiguration of the whitelist is going to prevent a block once an attack is known, nor is it going to make an attack more effective before it is. • We agree that we should minimize our use of the whitelist, and will be conducting a review/pruning of our current whitelist. • We are always looking to improve our security processes. We are actively hiring a chief security officer, who we hope to task with goals including optimizing our phishing detection process. If you know of anyone you'd recommend, please send them our way! • We will look into improving our visibility when the whitelist is modified. • We will investigate alternatives to our current phishing detection in the future. • We will look at PhishTank as a model for our phishing detection going forward."

[16] https[:]//www.bountysource[.]com/trackers/69336018-metamask-eth-phishing-detect
[17] https[:]//github[.]com/MetaMask/eth-phishing-detect/issues?utf8=%E2%9C%93&q=is%3Aissue+label%3A%22whitelist+request%22+
[18] https[:]//github[.]com/MetaMask/eth-phishing-detect/commit/d6148f41dadb5233818487189d870bc4af2ef847
[19] https[:]//crypto[.]bi/tape/blog/metamask/
[20] https[:]//www.phishtank.com
[21] https[:]//umbrella.cisco[.]com/blog/2017/11/27/introducing-phishtank_bot/
[22] https[:]//umbrella.cisco[.]com/blog/2017/09/27/protecting-icos-cryptocurrency-users/

Our Response	"Rolling your own" security technology and corresponding methodology should be carefully considered. Our stance is the publicly-available whitelist within the Chrome extension generates a valuable stamp of trust or approval by MetaMask, a well-recognized and respected project. This could lead threat actors to specifically compromise whitelisted infrastructure to avoid detection by users relying on the MetaMask phishing detector – and to potentially leverage this public "legitimacy" to additional third part(ies), increasing further downstream risk to users.

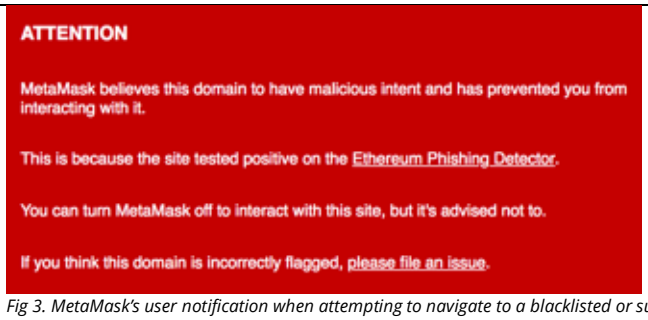
SG-MM-04. Stale Indicator Risk

Name	Stale Indicator Risk
Identifier	SG-MM-04
Description	Domain names (threat indicators) which have been added to the white- or blacklists aren't "aged" – meaning, there is no set expiration date at which point any given item will be reviewed or removed. Appropriate aging of indicators is a complex topic[23] for which there is no set industry standard. However, experts in threat intelligence agree there is a point past which the information is no longer useful to serve the goal of protecting users, false positives increase, and returns continually diminish. Domains and infrastructure which may have been clean when added to the whitelist may become controlled by a threat actor at a later date.
Recommendation	• Implement a review process or automatic removal based on age. • Develop a process for aging of indicators based on source and confidence.
Vendor Response	"We will look into automating an aging review process. We will consider weighing submission source when aging listed items."
Our Response	We thank MetaMask for consideration of these process changes and encourage the wider threat intelligence community to help analyze this issue.

SG-MM-05. Blocked Site Process Drives Improper User Behavior

Name	Blocked Site Process Drives Improper User Behavior
Identifier	SG-MM-05
Description	When a blacklisted or suspected phishing site is detected, users are directed to a live-hosted site[24] as shown in Figure 3 below which does not allow the user to continue navigating to the intended domain in spite of the warning.

[23] https[:]//www.threatq[.]com/expiration-entry-level-deprecation/
[24] https[:]//metamask[.]io/phishing.html

	 *Fig 3. MetaMask's user notification when attempting to navigate to a blacklisted or suspected phishing site.* • Users do not have a way to circumvent this warning and there is no way to bypass a false positive other than submit a complaint to MetaMask – and wait – or disable the extension. • As the complaint mechanism is based on Github and the majority of users will not have an account – especially as the use of crypto expands beyond enthusiasts – we posit this may incentivize the majority of users to either immediately disable the MetaMask extension to visit the desired domain or to register a new Github account to report the issue, thereby creating an inconvenience for the user and reducing the MetaMask team's ability to properly screen the individual making the request.
Recommendation	• Consider modifying the extension to allow users to navigate to the domain in spite of the warning. • Consider providing a user-friendly false positive submission process to reduce the likelihood the extension will be disabled and user security will be reduced as a result.
Vendor Response	"When users are unable to bypass the phishing page, they often come to us to complain, and at that time realize that the site was blocked for a reason. We will probably provide a mechanism to bypass the phishing warning in the future, but this is a low priority for us, because many users would click through without reading the warnings. We will be moving our report link to point at https[:]//etherscamdb[.]info/report instead of GitHub, to increase user friendliness. It feeds the same lists we draw from."
Our Response	We thank MetaMask for the changes and encourage the wider threat intelligence community to help analyze this issue.

SG-MM-06. Live Hosted "Blocked" Page Introduces Attack Vectors

Name	Live-Hosted Blocked Page
Identifier	SG-MM-06
Description	When a blacklisted or suspected phishing site is detected, users are directed to a live-hosted site[25] as shown in Finding SG-MM-05, Figure 3 above. This has multiple second-order implications: • Users attempting to visit a blocked site are directed to *metamask[.]io*, which creates a privacy leak for the user. However, this dataset can be a source of security data if intentionally collected. • *phishing.html* now becomes an attractive target for threat actors to compromise.
Recommendation	• Modify the code to direct the user to a locally hosted, static *phishing.html*, or ensure the live-hosted version is subject to robust change monitoring which alerts the team. • Consider intentionally collecting self-reported data from users regarding behavior leading to blocked page event to more deeply understand the security context generating the majority of events.
Vendor Response	"We will begin hosting our phishing.html page within the extension. We will consider incorporating an easy reporting feature within that page."
Our Response	We thank MetaMask for the changes and consideration, and encourage the wider threat intelligence community to help analyze this issue.

Conclusion

We appreciate and thank the MetaMask Team at ConsenSys for their work and support on our first public disclosure. Additional thanks to both our research partner Anomali[26] and to Twitter user @ninjininji[27], who performed an initial review of the "Blue Wallet[28]" Google Chrome extension code, chatted with us about it, and sparked a new interest in projects which reuse early code.

Our hope is to generate additional discussion and consideration of typical cybersecurity issues when implementing blockchain technologies. Thank you for reading this far and we look forward to your feedback on Twitter or Telegram[29]!

[25] https[:]//metamask[.]io/phishing.html
[26] https[:]//www.anomali[.]com
[27] https[:]//www.twitter[.]com/ninjininji
[28] https[:]//chrome.google[.]com/webstore/detail/blue-worlds-safest-simple/laphpbhjhhgigmjoflgcchgodbbclahk
[29] http[:]//www.cryptodefense[.]io