Least Authority
PRIVACY MATTERS

Bee + Bee Clef

Security Audit Report

# Swarm Association

Final Audit Report: 26 November 2021

# Table of Contents

*This audit makes no statements or warranties and is for discussion purposes only.*

# Overview

## Background

Swarm has requested that Least Authority perform a security audit of Bee and Bee-clef. Bee is a client implemented in Go and is the basic building block for Swarm Network. Bee-clef is a key manager for use with the Bee client and a custom instance of Go Ethereum's external signer, `clef`.

## Project Dates

- **April 28 - June 8**: Code review *(Completed)*
- **June 11**: Delivery of Initial Audit Report *(Completed)*
- **November 18 - 24:** Verification *(Completed)*
- **November 26:** Delivery of Final Audit Report *(Completed)*

## Review Team

- Dylan Lott, Security Researcher and Engineer
- Nathan Ginnever, Security Researcher and Engineer
- Phoebe Jenkins, Security Researcher and Engineer
- Rai Yang, Security Researcher and Engineer
- Steve Thakur, Cryptography Researcher and Engineer

# Coverage

## Target Code and Revision

For this audit, we performed research, investigation, and review of the Bee and Bee-clef followed by issue reporting, along with mitigation and remediation instructions outlined in this report.

The following code repositories are considered in-scope for the review:

- Bee: https://github.com/ethersphere/bee
- Bee-clef: https://github.com/ethersphere/bee-clef

Specifically, we examined the Git revisions for our initial review:

- Bee: `91af13c2578e5410d62b90592436fb533403705f`
- Bee-clef: `da82fabf3f4d7b6ec9fd1a838ab4dd19f09f15dc`

For the verification, we examined the Git revision:

- Bee: `175dfb4a5a56420d2cc1045cc7cc27ada2633373`
- Bee-clef: `f7a21f8b0e612fd7097485506ce0b69f25b2d77b`

For the review, these repositories were cloned for use during the audit and for reference in this report:

- https://github.com/LeastAuthority/Swarm-Bee
- https://github.com/LeastAuthority/Swarm-Bee-Clef

All file references in this document use Unix-style paths relative to the project's root directory.

In addition, any dependency and third party code, unless specifically mentioned as in-scope, were considered out of scope for this review.

## Supporting Documentation

The following documentation was available to the review team:

- Bee `README.md`: https://github.com/ethersphere/bee/blob/master/README.md
- Bee-clef `README.md`: https://github.com/ethersphere/bee-clef/blob/master/README.md
- Ethereum Swarm Medium: https://medium.com/ethereum-swarm
- Swarm Bee Documentation: https://docs.ethswarm.org/docs/
- V. Trón, "The Book of Swarm Storage and Communication Infrastructure for Self-Sovereign Digital Society Back-End Stack for the Decentralised Web." v1.0 pre-release 7 [T20]
- Decoupling Swap from Settlement: https://hackmd.io/@fu3G07ngSjOdUNbGVlew-w/SJKN0GjBu
- Swarm Network Layer Review (Draft): https://hackmd.io/9oPfcFTFSYupRlOVybGPZg?view
- Whitepaper Swarm (draft shared with Least Authority via Mattermost on 28 April 2021)

In addition, this audit report references the following documents:

- P. Maymounkov, D. Mazières, 2002, "Kademlia: A Peer-to-Peer Information System Based on the XOR Metric." *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, *Pages 53-65*. [PD02]

## Areas of Concern

Our investigation focused on the following areas:

- Correctness of the implementation;
- Common and case-specific implementation errors;
- Vulnerabilities in the client code, as well as secure interaction between the client and other network components;
- Key management: secure private key storage and proper management of encryption and signing keys;
- Exposure of API endpoints, which cause spending of funds;
- Attacks that impact funds, such as the draining or the manipulation of funds;
- Network attacks that impact client access to files and other data;
- Adversarial actions and protection against malicious attacks;
- Inappropriate permissions and excess authority;
- Data privacy, data leaking, and information integrity; and
- Anything else as identified during the initial analysis phase.

# Findings

## General Comments

Swarm is a decentralized storage and communication system aiming to serve a sovereign digital society. The system consists of peer-to-peer network nodes that create a decentralized storage and communication service. The system aims to be economically self-sustaining due to a built-in incentive system enforced through smart contracts on the Ethereum blockchain. The Bee client is an individual node in the system, representing the entry point to Swarm's data storage and distribution systems, and providing low level constructs for file storage, feeds, key-value stores, and untraceable communication. Bee-clef starts up a custom `clef` instance that is preconfigured to facilitate Bee's automated operation. Bee and Bee-clef were the core components in-scope for our security audit.

*This audit makes no statements or warranties and is for discussion purposes only.*

## System Design

We found that the design of Bee and Bee-clef strongly considers security. Below we outline aspects of the system design that we closely evaluated, in addition to our findings and conclusions.

### Use of Kademlia

Swarm uses a novel form of Kademlia [PD02] as implemented in Bee, referred to as Forwarding Kademlia. In Forwarding Kademlia, the lookup requests are forwarded through the network to an address closer to the requested ID, instead of being iteratively returned to the sender and the sender then starting a new lookup. While this difference is small, it is key to Swarm's system design. The ability for a message to pay for itself as a first-class citizen throughout the network via its postage stamp is an efficient way to incentivize message propagation, and provides spamming protection.

Message forwarding is a key part of the Swarm network and postage stamps allow messages to be handled by a node with zero trust in another actor. Swarm benefits from both the size and diversity of the network. Therefore, increasing message propagation and allowing nodes to profit off of forwarding incentivizes the network to maximize those properties. Forwarding Kademlia also delivers an optimization in average hop counts per any given message to its final destination. It does so by removing the iterative lookup that replies to the sending node at each hop that the classic Kademlia protocol describes. However, because it is not an iterative lookup, there are different attack vectors present in Forwarding Kademlia. As a result, Forwarding Kademlia is more susceptible to censorship and eclipse-based attacks.

We compared several different common attack vectors across Swarm's Forwarding Kademlia implementation and found problems and areas that warrant further investigation. In particular, we looked for eclipse attacks, blackhole attacks, sybil attacks, deanonymization attacks, and Denial of Service (DoS) attacks that could be carried out in Swarm. We found that eclipse (Issue B) and sybil (Issue A) attacks could lead to deanonymization, which creates the possibility of DoS attacks. However, this can be mitigated with techniques such as spoofless overlay addressing, proxy nodes, neighborhood depth upper limits, and having more nodes participating in the network. Blackhole attacks can be mitigated by incentivizing nodes to cache chunks outside of the neighborhood and resend the chunks if they are captured by the attackers (Issue F).

### Identity and Anonymity

Bee's key assumptions focus on identity and anonymity. Given Swarm's reliance on shielding identity, preserving anonymity, and plausible deniability as a participation incentive, deanonymization attacks create a disproportionately large vulnerability for Swarm, necessitating additional precautions around identity protection and anonymity preservation. For example, any deanonymizing attack would allow a node to be targeted by a DoS attack or a censorship attack. In examining the implementation for different methods that could result in the extraction of identity information from the system as it operates, we identified two issues (Issue A, Issue B) that we consider deanonymization vectors.

### Chunk Size

Swarm's 4k static chunk size allows for benefits including simplification of the protocol and anonymization of file size. However, the 4k static chunk size may also prove to be a scaling bottleneck. While this size adds negligible overhead for small files and messaging through feeds, if a user was retrieving a large file on the order of hundreds of megabytes, significant additional overhead traffic would be generated.

The relatively small and static chunk size is creatively engineered into the protocol. Since chunks are encrypted, fixed chunk sizes obscure the network's traffic even if all of the encrypted traffic is visible, which provides additional security against deanonymization and targeting or censorship.

*This audit makes no statements or warranties and is for discussion purposes only.*

### PSS and Feed

Swarm builds their entire protocol on top of the chunk primitive they have created and, at every level of the protocol, they have wrapped it differently to accommodate the needs of specific packages. PSS messaging is the messaging system in which a message is built on a special kind of chunk called a Trojan chunk. Feeds are a content updating (pub-sub) system with numerous novel use cases, including, but not limited to, message broadcasting, group chat, and content updates streaming. Feeds are based on a special chunk called a single owner chunk. This is a practical and efficient way to use an existing framework for communication in a variety of ways, while requiring a minimal amount of new code.

Furthermore, Trojan chunks provide a way to anonymize traffic of different types as they are encrypted and have the same size as normal chunks, making them indistinguishable from the normal file chunk for an attacker, except to the recipient. The underlying peer-to-peer communication protocol does not change, but they are able to use it to accomplish multiple composite features, from real time feed updates to private feeds with end-to-end encryption.

We found the PSS and feed chunks system enhance the security of the system and is an efficient way to facilitate both one-to-one direct and one-to-many broadcast communication without exposing nodes to unnecessary risk or denial of service attacks.

### Swap Vulnerability

Swap is built to incentivize bandwidth sharing while keeping a zero-cost entry threshold for the network. One of the key trade offs that every decentralized storage protocol has to face is supply versus demand side limitations. If the cost to participate in the network is zero, the barrier to entry is substantially minimized and can increase adoption, but also increases the opportunity for abuse. Conversely, a non-zero entry cost removes this ability to abuse the network but hinders wide adoption, requiring Swarm to heavily incentivize adoption in order to both secure their own network, and to keep the network large enough to maintain its underlying security guarantees. This Tragedy of the Commons problem frequently surfaces in distributed storage protocols. We recommend that Swarm continue to carefully consider this problem and rigorously test any changes that interact with the Swap accounting system.

### Storage Incentive

#### Postage Stamp

To protect against spamming attacks and incentivize chunk persistence, Swarm uses postage stamps to charge for chunk uploading. The uploader has to purchase a postage stamp upfront and attach the stamp to the chunk before it can be uploaded. The cost of uploading a chunk also serves as a price signal of the relative importance of the content. Storer nodes can then use the cost of uploading to rank chunks when selecting which ones to retain and serve and which to garbage collect in the event that storage capacity is reached. This solution promotes security best practices and does not rely on a third party for trust, since any forwarding node can see and check the balance of the postage stamp and thus has zero risk for handling a message. We consider this a secure and efficient way of handling messages in the Swarm ecosystem.

#### Postage Lottery

One of the main incentives described in the Book of Swarm [T20] is the postage lottery. Storer nodes are rewarded by a probabilistic payment for both being online and responsive at any given time, which a randomly chosen storer node demonstrates with proofs in response to generated challenges. This is an efficient and fair way to reward good behavior while minimizing on-chain transaction costs in the long term. In Kademlia, uptime is a crucial indicator of a node's future likelihood to be online [T20].

The postage lottery incentivizes good uptime behavior while simultaneously allowing for small periods of downtime for hardware and software updates, random outages, and other unplanned events. By committing to a storage lottery round, a reasonable assumption can be made that the same node will do the same thing again in the next round. However, this component of Swarm is not yet implemented in Bee. We recommend a full audit of this system once it has been fully implemented in Bee (Suggestion 3).

### Bee-clef

We examined Bee-clef, the Ethereum key manager tailored for Bee, and did not identify any security issues. Users should be advised against using HTTP ports with the Bee-clef API. In the event that  HTTP is configured, a firewall should be used. Bee-clef sets reasonable default configurations for Bee to use, which we generally encourage for security sensitive applications. It should be noted however, that because it simply wraps Clef, the Ethereum key manager, we cannot comment on the security of the core Clef code. We recommend that a follow up security audit be conducted on the `clef` implementation (Suggestion 6).

## Fuzz Testing

In addition to our manual code review of the code, we examined marshaling and unmarshaling logic around chunk creation and validation. In particular, we searched for any functions that would cover a significant amount of code but abstract it away and wrote fuzz testing functions and ran them both with and without a corpus. We did not identify any crashing inputs that caused security issues or unrecoverable panics.

## Code Quality + Documentation

The codebase is substantial and considerably complex, however, it is structured such that it is easy to navigate. We found Bee and Bee-clef to be well documented and sufficiently commented. Bee implements a robust test suite with a commendable use of a custom mocking solution for testing that allows for very precise tests to be written for any component of the system that provides a mocking package for itself. Sufficient documentation and test coverage provides an additional layer of security by enabling both reviewers and contributors of the project to more easily understand the intended behavior of the system, thus facilitating the ability to identify potential vulnerabilities, bugs, errors, and edge cases.

Our team had some initial difficulty determining what was and was not implemented in our locked commit version of the Bee and Bee-clef repositories due to some references to specific behaviors and implementation details being outdated. The Swarm team was helpful in providing clarifications and made themselves available for questions throughout the duration of the audit. However, we recommend that all documentation be updated to reflect the current status of the implementation (Suggestion 5).

## Scope & Dependencies

We found the scope of the audit for Bee to be sufficient, in that it encompassed all security critical components for that implementation. The scope for the audit of Bee-clef was insufficient in that clef was out of scope (Suggestion 6). Swarm is built with libp2p as the underlying network library, which is a battle tested framework that has been audited and has widespread community support.

In the future, we recommend follow up security audits for more targeted subsystems of the Swarm protocol. The Swap, Swear, and Swindle smart contracts that power Swarm were out of scope for this audit, and are a security critical layer of the network that abstracts away significant complexity. We recommend a full audit of the smart contracts, with particular focus on Bee's interaction with the smart contracts (Suggestion 2).

In addition, we recommend an audit of the postage lottery system once it has been implemented (Suggestion 3).

*This audit makes no statements or warranties and is for discussion purposes only.*

## Specific Issues & Suggestions

We list the issues and suggestions found during the review, in the order we reported them. In most cases, remediation of an issue is preferable, but mitigation is suggested as another option for cases where a trade-off could be required.

| ISSUE / SUGGESTION | STATUS |
|---|---|
| Issue A: Deanonymization Attack Targeting Sender Node of Proximity Order 0 and Light Node | Unresolved |
| Issue B: Deanonymization Attack Based on Eclipse Attack | Partially Resolved |
| Issue C: Incorrect gcSizeChange Updates | Resolved |
| Issue D: Access i.PinCounter Without Checking for Error | Resolved |
| Issue E: Wrong Tier When Postage Batch value=0 | Resolved |
| Issue F: Blackhole - (Censorship) Attacks | Unresolved |
| Issue G: Batches With a Very Small Batch Value May Enter the Batch Store | Partially Resolved |
| Suggestion 1: Remove Redundant Code | Resolved |
| Suggestion 2: Conduct a Security Audit of Smart Contracts and Integration with Bee | Unresolved |
| Suggestion 3: Conduct a Security Audit of Postage Lottery Systems | Unresolved |
| Suggestion 4: Update Docker Base Image | Resolved |
| Suggestion 5: Update Documentation | Resolved |
| Suggestion 6: Conduct a Security Audit of `clef` Implementation | Unresolved |

### Issue A: Deanonymization Attack Targeting Sender Node of Proximity Order 0 and Light Node

**Synopsis**

In Swarm, sender anonymity, or more precisely, the ambiguity of a node, is based on Forwarding Kademlia routing, thus a node can be a forwarder or an originator of a chunk or message. However, there are two exceptions:

1. If the chunk or PSS message is sent from a node with Proximity Order (PO) 0, the node must be an originator of the chunk/message.
2. Light node can only be the originator as light node can not forward chunk/message.

*This audit makes no statements or warranties and is for discussion purposes only.*

### Impact

Nodes with PO 0 or a light node would be an easy target for a deanonymization attack, which could result in leaks of network meta data (e.g. IP, location) through Kademlia DHT. This network metadata could enable other attacks including DoS attacks and censorship attacks. However, this deanonymization attack would not reveal the contents of the chunk/message.

### Preconditions

A chunk/PSS message is sent by a node of PO 0 or a light node to the attacker node.

### Feasibility

The feasibility of this attack depends on the number of attacker nodes. For every attacker node, there is a 50% percent chance of being the PO 0 node. The probability of a successful attack is linear in the number of attacker nodes.

### Mitigation

The following are potential approaches to mitigate this issue:

- Use a proxy node to hide the sender node. The limitation of this approach is that if the proxy node is attacked, the sender node will be deanonymized or taken down.
- A Bee node owner can use network layer anonymization tools such as Onion routing, mixnet, or others to connect to the Swarm network.
- The planned spoof-resistant overlay addressing to mitigate neighborhood mining will also reduce the feasibility of an attacker generating a large number of malicious nodes.
- A more comprehensive mitigation to this class of deanonymization attacks is to attract as many nodes to the network as possible, thus reducing the chance of sybil attacks leading to deanonymization attacks.

We recommend that the Swarm team continue to improve the incentive mechanism, including Swap and postage lottery, in order to attract more users to the network and increase overall security. In addition, we suggest advising users of the risk of this attack and the mitigation approaches.

### Status

The Swarm team has responded that they have not yet identified an effective solution to resolve this issue. We recommend that the Swarm team continue to explore effective mitigation strategies to improve the incentive mechanism.

### Verification

Unresolved.

## Issue B: Deanonymization Attack Based on Eclipse Attack

### Synopsis

In the case that a victim node is eclipsed in a neighborhood and if the victim node is the sender/retriever/receiver of a chunk or message, it will be deanonymized by the attacker.

### Impact

Once a node is deanonymized, the node becomes vulnerable to other attacks such as DoS attacks, censorship attacks, and others.

*This audit makes no statements or warranties and is for discussion purposes only.*

### Preconditions

The attacker nodes occupy the neighborhood of the victim node in an eclipse attack by mining addresses in the neighborhood and capturing the chunks sent to or from the victim node.

### Feasibility

The probability of capturing a neighborhood is low, as there is an upper bound of neighborhood depth. Although currently not implemented, spoofless address features that may be introduced in the future could make mining addresses in a specific neighborhood prohibitively difficult and expensive.

### Technical Details

Attacker nodes occupy the neighborhood; any chunk/message sent to or from the victim node will be captured by the attacker, thus capturing the sender overlay address and other network meta data such as IP address and location. This data leak could result in other kinds of attacks such as DoS, censorship attacks, and others.

### Remediation

We recommend taking steps to make address mining costly as a mitigation for eclipse attacks. We note that spoofless addresses are proposed for implementation. This step would make address mining costly, although Ethereum miners can still mine addresses by manipulating the block hash.

We also note that the neighborhood depth upper limit makes occupying a neighborhood difficult. Additionally, the chunks/messages content is encrypted, therefore the content's link to its sender and receiver would not be revealed. Furthermore, it is very difficult for an attacker to distinguish a normal chunk from a PSS message (a Trojan chunk), further obfuscating the message.

### Status

The Swarm team has implemented spoofless overlay network addresses, which use the Ethereum blockhash (one block after the block that contains the node's chequebook transaction) as a source of randomness for overlay addresses, which strongly discourages address mining. Theoretically, however, Ethereum miners can still be bribed to mine addresses in a specific neighborhood by filtering addresses which do not lie in the victim node's neighborhood. The current implemented mitigation makes that action prohibitively expensive and difficult to control, as it is difficult for the miner to mine two consecutive blocks. However, in order to fully resolve the issue, the Swarm team needs to make address mining impossible by implementing a better source of randomness for address derivation (e.g. beacon chain RANDAO in Eth 2.0), which is not available in Eth 1.0. Thus, we recommend that the Swarm team continue to explore new options as they become available to fully resolve the issue.

### Verification

Partially Resolved.

## Issue C: Incorrect gcSizeChange Updates

### Location

`pkg/localstore/mode_put.go#L136`

### Synopsis

When storing the chunk to local storage, garbage collection size is updated. However, it is updated twice in the wrong location.

### Impact

The garbage collection size will update to an incorrect size, resulting in distorted garbage collecting behavior.

### Preconditions

The chunk's upload mode is set to `ModePutUpload`.

### Feasibility

Straightforward.

### Technical Details

When chunk's upload mode is set to `ModePutUpload`, `gcSizeChange` will be updated twice with

`gcSizeChange += c`.

### Remediation

We recommend moving the second `gcSizeChange` update into the `mode == storage.ModePutUploadPin` branch case.

```
 if mode == storage.ModePutUploadPin {
      c, err = db.setPin(batch, item)
      if err != nil {
        return nil, err
      }
      gcSizeChange += c
}
```

### Status

The Swarm team has implemented a remediation for the `gcSizeChange` update, as suggested.

### Verification

Resolved.

## Issue D: Access i.PinCounter Without Checking for Error

### Location

`pkg/localstore/mode_set.go#L268`

### Synopsis

When a chunk is pinned to be excluded from garbage collection, the existing pin counter needs to be fetched from the `pin index`. The error from fetch is not checked before accessing the `pin counter`.

### Impact

The program will panic and exit if there is an error from fetching the `pin index(pinIndex)`.

*This audit makes no statements or warranties and is for discussion purposes only.*

**Preconditions**

There is an error (e.g. address not found) in the execution of fetch on the pin counter.

**Feasibility**

Straightforward.

**Technical Details**

In function `setPin`:

```
          i, err := db.pinIndex.Get(item)
                      item.PinCounter = i.PinCounter
          if err != nil {
                    …
```

`i.PinCounter` is accessed before the error check. If there is an error, `i.PinCounter` would panic the program and exit.

**Remediation**

We recommend moving `item.PinCounter = i.PinCounter` down after the error check block.

**Status**

The Swarm team has responded and provided supporting evidence that this is a non-issue. In particular, `shed.Index.Get` returns an instance, not a pointer. Rather than panic on error, the default value of the pin counter on the item (zero) is used instead. We acknowledge and confirm that this issue is no longer valid.

**Verification**

Resolved.

## Issue E: Wrong Tier When Postage Batch value=0

**Location**

`postage/batchstore/reserve.go#L193`

**Synopsis**

When the postage batch value=0, the tier function returns the wrong tier (inner tier) instead of `unreserved`.

**Impact**

The wrong tier results in a wrong reserve size calculation, enabling an incorrect chunk allocation in the reserve.

**Preconditions**

The value of a postage batch is 0.

**Technical Details**

In the tier function, the tier is returned based on postage value compared to inner value and outer value of a reserve:

```
  // x < rs.Inner || x == 0

   if x.Cmp(rs.Inner) < 0 || rs.Inner.Cmp(big.NewInt(0)) == 0 {
```

```
            return unreserved
    }


 if x.Cmp(rs.Outer) < 0 {
       return inner
 }
```

If x == 0,  tier would return inner value instead of unreserved (the correct tier).

### Remediation

We recommend replacing

```
rs.Inner.Cmp(big.NewInt(0)) == 0
```

with

```
x.Cmp(big.NewInt(0)) == 0.
```

### Status

The Swarm team has responded that the batch value is normalized value equal to cumulative payout per chunk plus the amount per chunk approved by the postage stamp smart contract, which will always be greater than 0 (except at epoch 0). As a result, we acknowledge and confirm that this issue is no longer valid. However, a related issue with very small batch value was identified while investigating this issue (see Issue G).

### Verification

Resolved.

## Issue F: Blackhole (Censorship) Attacks

### Location

pkg/pushsync/pushsync.go#L256

pkg/pushsync/pushsync.go#L269

### Synopsis

A malicious/censoring forwarder node can forge a receipt message that says that a chunk was stored in its destination when in reality it was dropped or ignored.

### Impact

This could lead to lost chunks in the network and an attacker node gaining an illegitimate reward.

### Preconditions

A victim node must send a chunk to a destination, which is dropped by a malicious forwarder node in a forwarding route.

*This audit makes no statements or warranties and is for discussion purposes only.*

### Feasibility

Moderate. A forwarder node is incentivised not to forward the chunk and claim the reward. Alternatively, a single owner chunk could be censored if somehow the attacker knows it's a feed chunk, even without incentive.

### Technical Details

An attacker node receives a chunk from another node. The attacking node drops or discards the chunk and passes back a forged receipt indicating that the chunk was stored successfully. The attacking node would then be eligible to claim a reward via the SWAP protocol.

### Remediation

We recommend implementing a change where the uploader and forwarder (who caches the chunk incentivised by SWAP) can verify uploading/forwarding (push-sync) by retrieving a chunk through an alternative route after a successful forwarding within a random amount of time. If the retrieval fails, the node can push/sync the cached chunk through an alternative route.

### Status

The Swarm team has responded that implementing the suggested remediation is non-trivial due to the cost and the significant required change to the architecture, in addition to making syncing slow and unreliable. They note that guaranteeing disjoint path traversal for retrieval and push is complex and difficult to achieve. In addition, the Swarm team does not want to introduce cross protocol dependency (relying on the forwarder) as it would introduce additional complexity. Instead, they are working towards identifying and implementing solutions that would result in a more reliable syncing process that is less susceptible to manipulation (i.e. the uploader verifies the upload and resends chunk via a disjoint path if the verification fails). We acknowledge their decision and agree that the proposed solution would be sufficient, as long as the disjoint path is not blocked by a blackhole attack. Our suggested remediation is an extension of this approach. Given that a solution has yet to be implemented and verified by our team at this time, this issue remains unresolved.

### Verification

Unresolved.

## Issue G: Batches With a Small Batch Value May Enter the Batch Store

### Location

`postage/batchservice/batchservice.go#L33`

`master/src/PostageStamp.sol#L109`

### Synopsis

A batch with a small batch value might enter the batch store, since there is no zero or  amount per chunk value check in the postage stamp contract, causing other valid batches to be marked (queued) for eviction once the node's reserve capacity is reached.

### Impact

Valid batches will be marked (queued) for eviction when the node's reserve capacity is reached.

### Preconditions

A batch with a zero or very small balance per chunk is purchased and created in the postage stamp contract.

*This audit makes no statements or warranties and is for discussion purposes only.*

#### Technical Details

As batch value is normalized value equal to cumulative payout per chunk plus amount per chunk approved by the postage stamp contract, since there is no zero or small amount per chunk value check in the postage stamp smart contract so a batch with a very small  amount per chunk value might enter the batch store, increasing the reserve radius. This would result in other valid batches being marked (queued) for eviction when the node's reserve capacity is reached. The newly added small value batch will also be marked for eviction in the subsequent blocks. Once it is evicted, the reserve radius has to decrease again, and the previous queued valid batches will be dequeued (unmarked) from being evicted (this has not been implemented and the Swarm team has indicated that they are actively working on it).

#### Mitigation

We recommend not adding batches that have a value equaling total cumulative payout (amount per chunk =0) or batches that are going to expire (small amount per chunk value) within the next couple of blocks.

#### Remediation

We recommend allowing the batches with a zero or small amount per chunk value to enter the batch store, which increases the reserve radius. However, reserve radius should decrease once the batch with small values is evicted, and previous queued valid batches should be dequeued (unmarked) from being evicted.

#### Status

The Swarm team has applied a stop gap to prevent adding batches that have a value equaling total cumulative payout or batches that are going to expire within the next couple of blocks. The mitigation has been implemented and merged with the master branch. In addition, the postage stamp smart contract will also be updated to block the 0 value batch(`_initialBalancePerChunk = 0`) in the next iteration.

Our team agrees that this is an effective mitigation as long as the price event on-chain can be accurately tracked (which is updated every ½ block), the price is not updated too frequently (it is currently updated manually once the network capacity is reached), and is least intrusive to the existing nodes. Given that a full remediative is still in progress, this issue remains partially resolved at the time of this verification.

#### Verification

Partially Resolved.

# Suggestions

## Suggestion 1: Remove Redundant Code

#### Location

`pkg/localstore/mode_set.go#L195-L199`

`pkg/localstore/mode_put.go#L323-L329`

#### Synopsis

The code for the population of `Item` with radius data is redundant here, as it is already in the `preserveOrCache` function below. Redundant code inhibits understanding the intended functionality of the codebase for developers and security researchers.

*This audit makes no statements or warranties and is for discussion purposes only.*

**Mitigation**

We suggest that the codebase be scanned for redundant code, followed by the removal of all instances of redundant code. For example, remove the redundant code between L195-L199.

**Status**

The Swarm team has removed the redundant code.

In addition, code has also undergone significant restructuring since the delivery of the Initial Audit Report. The restructured code has not been reviewed by our team as part of this verification.

**Verification**

Resolved.

## Suggestion 2: Conduct a Security Audit of Smart Contracts and Integration with Bee

**Location**

`ethersphere/swap-swear-and-swindle`

**Synopsis**

The Swap, Swear, and Swindle smart contracts were not in-scope for our audit and are a security critical layer of the network that abstracts away significant complexity.

**Mitigation**

We recommend a full audit of the Swap, Swear, and Swindle smart contracts that power Swarm and the way in which they interact with Bee.

**Status**

The Swarm team has responded that an audit of the Swap, Swear, and Swindle smart contracts has not yet been completed.

**Verification**

Unresolved.

## Suggestion 3: Conduct a Security Audit of Postage Lottery Systems

**Location**

Section 3.3.2 in the Book of Swarm

**Synopsis**

The Book of Swarm describes a postage lottery process (Raffle, Apply, Claim, and Earn, or RACE) for rewarding nodes probabilistically in the long term, which can successfully respond to a challenge proving that they still own a chunk. This system is not implemented in Bee. Because this is such an integral part of the Swarm protocol, we recommend a full audit of this feature once it has been fully implemented.

**Mitigation**

We recommend conducting an audit of the postage lottery system once it has been implemented.

*This audit makes no statements or warranties and is for discussion purposes only.*

**Status**

The Swarm team has responded that this feature has not yet been implemented. Once it is implemented, the Swarm team has stated that they intend to have it reviewed by an independent security auditing team.

**Verification**

Unresolved.

## Suggestion 4: Update Docker Base Image

**Location**

`bee/Dockerfile`
`bee/Dockerfile.goreleaser`

**Synopsis**

Docker's scanning and security tool revealed some issues related to an out-of-date base Docker image that is no longer maintained and contains known security vulnerabilities. This can be fixed with a base image update. Some of the issues discovered included out-of-bound writes, insufficient randomness, and information leakage. This is a Docker security issue, so it is harder to exploit than a network level issue, but it is still possible that deanonymization could lead to one of these vectors being exploited.

**Mitigation**

We recommend updating Bee's Docker files to use a newer, currently maintained base version of the `alpine` image that contains the relevant security patches.

**Status**

The Swarm team has updated the Dockerfile to build with the maintained version of Debian.

**Verification**

Resolved.

## Suggestion 5: Update Documentation

**Location**

The Book of Swarm [T20]

Supporting Documentation

**Synopsis**

Several areas of the documentation were missing, blank, or out-of-date with the implementation in-scope for the security audit. Additionally, the project documentation does not make immediately clear which parts of the system design have been implemented or are still pending implementation.

**Mitigation**

We recommend the Swarm team update the project documentation to match what is expected to be in the code for future audits.

**Status**

The Swarm team has updated the documentation. However, given that development has continued since the delivery of the Initial Audit Report, we have not verified that the documentation is consistent with the

*This audit makes no statements or warranties and is for discussion purposes only.*

current implementation. As a result, we recommend that the Swarm team make continuous updates to the documentation in accordance with the implementation.

**Verification**

Resolved.

## Suggestion 6: Conduct a Security Audit of `clef` Implementation

**Location**

`master/cmd/clef`

**Synopsis**

The `clef` implementation was not in-scope for our audit and, to the best of our knowledge, has not undergone a security audit since 2018.

**Mitigation**

We recommend conducting an audit of the `clef` implementation, as it has critical implications on the security of the Bee implementation.

**Status**

The Swarm team has responded that an audit of the `clef` implementation has not yet been completed.

**Verification**

Unresolved.

# Appendix 1 - Docker Scan Results

## Bee Docker Scan Results

```
> $ docker scan ethersphere/swarm
Testing ethersphere/swarm…
✗ Low severity vulnerability found in openssl/libcrypto1.1

  Description: Inadequate Encryption Strength

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089236

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1j-r0


✗ Low severity vulnerability found in openssl/libcrypto1.1

  Description: Missing Encryption of Sensitive Data

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-505098

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1d-r0
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
✗ Medium severity vulnerability found in openssl/libcrypto1.1

  Description: NULL Pointer Dereference

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089231

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1k-r0


✗ Medium severity vulnerability found in openssl/libcrypto1.1

  Description: NULL Pointer Dereference

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089233

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1i-r0


✗ Medium severity vulnerability found in openssl/libcrypto1.1
```

```
  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089234

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1j-r0


✗ Medium severity vulnerability found in openssl/libcrypto1.1

  Description: Missing Encryption of Sensitive Data

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-491992

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1d-r0


✗ Medium severity vulnerability found in openssl/libcrypto1.1

  Description: Use of Insufficiently Random Values

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-501158
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1d-r0



✗ Medium severity vulnerability found in openssl/libcrypto1.1

  Description: Information Exposure

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-588019

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1d-r2



✗ Medium severity vulnerability found in musl/musl

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-ALPINE39-MUSL-1042761

  Introduced through: musl/musl@1.1.20-r4, busybox/busybox@1.29.3-r10,
alpine-baselayout/alpine-baselayout@3.1.0-r3, openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, zlib/zlib@1.2.11-r1,
apk-tools/apk-tools@2.10.3-r1, libtls-standalone/libtls-standalone@2.7.4-r6,
busybox/ssl_client@1.29.3-r10, ca-certificates/ca-certificates@20190108-r0,
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
musl/musl-utils@1.1.20-r4, pax-utils/scanelf@1.2.3-r0,
libc-dev/libc-utils@0.7.1-r0

  From: musl/musl@1.1.20-r4

  From: busybox/busybox@1.29.3-r10 > musl/musl@1.1.20-r4

  From: alpine-baselayout/alpine-baselayout@3.1.0-r3 > musl/musl@1.1.20-r4

  and 11 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.20-r6


✗ High severity vulnerability found in openssl/libcrypto1.1

  Description: Improper Certificate Validation

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089232

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1k-r0


✗ High severity vulnerability found in openssl/libcrypto1.1

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-1089235

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1
  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1j-r0



✗ High severity vulnerability found in openssl/libcrypto1.1

  Description: NULL Pointer Dereference

  Info: https://snyk.io/vuln/SNYK-ALPINE39-OPENSSL-588029

  Introduced through: openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, apk-tools/apk-tools@2.10.3-r1,
libtls-standalone/libtls-standalone@2.7.4-r6,
ca-certificates/ca-certificates@20190108-r0

  From: openssl/libcrypto1.1@1.1.1b-r1

  From: openssl/libssl1.1@1.1.1b-r1 > openssl/libcrypto1.1@1.1.1b-r1

  From: apk-tools/apk-tools@2.10.3-r1 > openssl/libcrypto1.1@1.1.1b-r1

  and 5 more...

  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'

  Fixed in: 1.1.1g-r0



✗ High severity vulnerability found in musl/musl

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-ALPINE39-MUSL-458529

  Introduced through: musl/musl@1.1.20-r4, busybox/busybox@1.29.3-r10,
alpine-baselayout/alpine-baselayout@3.1.0-r3, openssl/libcrypto1.1@1.1.1b-r1,
openssl/libssl1.1@1.1.1b-r1, zlib/zlib@1.2.11-r1,
apk-tools/apk-tools@2.10.3-r1, libtls-standalone/libtls-standalone@2.7.4-r6,
busybox/ssl_client@1.29.3-r10, ca-certificates/ca-certificates@20190108-r0,
musl/musl-utils@1.1.20-r4, pax-utils/scanelf@1.2.3-r0,
libc-dev/libc-utils@0.7.1-r0

  From: musl/musl@1.1.20-r4

  From: busybox/busybox@1.29.3-r10 > musl/musl@1.1.20-r4
  From: alpine-baselayout/alpine-baselayout@3.1.0-r3 > musl/musl@1.1.20-r4
  and 11 more...
  Image layer: '/bin/sh -c apk --no-cache add ca-certificates &&
update-ca-certificates'
  Fixed in: 1.1.20-r5
Project name:      docker-image|ethersphere/swarm
Docker image:      ethersphere/swarm
Platform:          linux/amd64
Licenses:          enabled
Tested 15 dependencies for known issues, found 13 issues.
```

## Bee Clef Docker Scan Results

```
> $ docker scan ethersphere/clef:0
Testing ethersphere/clef:0...
✗ Low severity vulnerability found in tar
  Description: Out-of-bounds Read
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-1063001
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > tar@1.30+dfsg-6

✗ Low severity vulnerability found in tar
  Description: CVE-2005-2541
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-312331
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > tar@1.30+dfsg-6

✗ Low severity vulnerability found in tar
  Description: NULL Pointer Dereference
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-341203
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > tar@1.30+dfsg-6


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Authentication Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-1291056

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Link Following

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-305144

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Missing Release of Resource after Effective Lifetime

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-542807

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2
```

```
  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Improper Input Validation

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-570991

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Time-of-check Time-of-use (TOCTOU)

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306205

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Incorrect Permission Assignment for Critical Resource

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306230
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Access Restriction Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306250

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Incorrect Permission Assignment for Critical Resource

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-539852

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in perl/perl-base

  Description: Link Following

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-327793
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345321
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345353
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Uncontrolled Recursion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345502
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345530
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-bounds Read
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-572368

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12


✗ Low severity vulnerability found in nettle/libnettle6

  Description: CVE-2021-3580

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-NETTLE-1301269

  Introduced through: nettle/libnettle6@3.4.1-1, apt@1.8.2,
nettle/libhogweed4@3.4.1-1

  From: nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libhogweed4@3.4.1-1
> nettle/libnettle6@3.4.1-1

  and 2 more...


✗ Low severity vulnerability found in lz4/liblz4-1

  Description: CVE-2021-3520

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LZ4-1277601

  Introduced through: lz4/liblz4-1@1.8.3-1, apt@1.8.2

  From: lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > lz4/liblz4-1@1.8.3-1

  Fixed in: 1.8.3-1+deb10u1


✗ Low severity vulnerability found in lz4/liblz4-1

  Description: Buffer Overflow

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LZ4-473072

  Introduced through: lz4/liblz4-1@1.8.3-1, apt@1.8.2

  From: lz4/liblz4-1@1.8.3-1
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > lz4/liblz4-1@1.8.3-1


✗ Low severity vulnerability found in libtasn1-6

  Description: Resource Management Errors

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBTASN16-339585

  Introduced through: libtasn1-6@4.13-3, apt@1.8.2

  From: libtasn1-6@4.13-3

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libtasn1-6@4.13-3


✗ Low severity vulnerability found in libseccomp/libseccomp2

  Description: Access Restriction Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBSECCOMP-341044

  Introduced through: libseccomp/libseccomp2@2.3.3-4, apt@1.8.2

  From: libseccomp/libseccomp2@2.3.3-4

  From: apt@1.8.2 > libseccomp/libseccomp2@2.3.3-4


✗ Low severity vulnerability found in libgcrypt20

  Description: CVE-2021-33560

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-1297893

  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2

  From: libgcrypt20@1.8.4-5

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5


✗ Low severity vulnerability found in libgcrypt20

  Description: Use of a Broken or Risky Cryptographic Algorithm

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-391902
  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2
  From: libgcrypt20@1.8.4-5
  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5

✗ Low severity vulnerability found in gnutls28/libgnutls30
  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-340755
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ Low severity vulnerability found in gnupg2/gpgv
  Description: Use of a Broken or Risky Cryptographic Algorithm
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUPG2-535553
  Introduced through: gnupg2/gpgv@2.2.12-1+deb10u1, apt@1.8.2
  From: gnupg2/gpgv@2.2.12-1+deb10u1
  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Double Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1078993
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Uncontrolled Recursion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-338106
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Uncontrolled Recursion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-338163
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356371
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Resource Management Errors
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356671
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Resource Management Errors
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356735
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: CVE-2010-4051

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356875

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Out-of-Bounds

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-452228

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Access Restriction Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-452267

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Use of Insufficiently Random Values

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-453375

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Information Exposure

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-453640

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Information Exposure

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-534995

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Integer Underflow

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-564233

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in coreutils

  Description: Improper Input Validation

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-COREUTILS-317465

  Introduced through: coreutils@8.30-3

  From: coreutils@8.30-3
```

```
✗ Low severit> $ docker scan ethersphere/clef:0
⬡ 15.3.0


Testing ethersphere/clef:0...


✗ Low severity vulnerability found in tar

  Description: Out-of-bounds Read

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-1063001

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > tar@1.30+dfsg-6


✗ Low severity vulnerability found in tar

  Description: CVE-2005-2541

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-312331

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > tar@1.30+dfsg-6


✗ Low severity vulnerability found in tar

  Description: NULL Pointer Dereference

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-TAR-341203

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > tar@1.30+dfsg-6


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Authentication Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-1291056

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Link Following

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-305144

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Missing Release of Resource after Effective Lifetime

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-542807

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in systemd/libsystemd0

  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-570991

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Time-of-check Time-of-use (TOCTOU)

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306205

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Incorrect Permission Assignment for Critical Resource

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306230

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd
```

```
  Description: Access Restriction Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-306250

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in shadow/passwd

  Description: Incorrect Permission Assignment for Critical Resource

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SHADOW-539852

  Introduced through: shadow/passwd@1:4.5-1.1, adduser@3.118,
shadow/login@1:4.5-1.1, util-linux/mount@2.33.1-0.1

  From: shadow/passwd@1:4.5-1.1

  From: adduser@3.118 > shadow/passwd@1:4.5-1.1

  From: shadow/login@1:4.5-1.1

  and 1 more...


✗ Low severity vulnerability found in perl/perl-base

  Description: Link Following

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-327793

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6


✗ Low severity vulnerability found in pcre3/libpcre3

  Description: Out-of-Bounds

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345321

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12
```

```
✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345353
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Uncontrolled Recursion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345502
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-345530
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in pcre3/libpcre3
  Description: Out-of-bounds Read
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-572368
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Low severity vulnerability found in nettle/libnettle6
  Description: CVE-2021-3580
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-NETTLE-1301269
  Introduced through: nettle/libnettle6@3.4.1-1, apt@1.8.2,
nettle/libhogweed4@3.4.1-1

  From: nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libhogweed4@3.4.1-1
> nettle/libnettle6@3.4.1-1

  and 2 more...


✗ Low severity vulnerability found in lz4/liblz4-1

  Description: CVE-2021-3520

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LZ4-1277601

  Introduced through: lz4/liblz4-1@1.8.3-1, apt@1.8.2

  From: lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > lz4/liblz4-1@1.8.3-1

  Fixed in: 1.8.3-1+deb10u1


✗ Low severity vulnerability found in lz4/liblz4-1

  Description: Buffer Overflow

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LZ4-473072

  Introduced through: lz4/liblz4-1@1.8.3-1, apt@1.8.2

  From: lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > lz4/liblz4-1@1.8.3-1

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > lz4/liblz4-1@1.8.3-1


✗ Low severity vulnerability found in libtasn1-6

  Description: Resource Management Errors

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBTASN16-339585

  Introduced through: libtasn1-6@4.13-3, apt@1.8.2
  From: libtasn1-6@4.13-3

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libtasn1-6@4.13-3


✗ Low severity vulnerability found in libseccomp/libseccomp2

  Description: Access Restriction Bypass

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBSECCOMP-341044

  Introduced through: libseccomp/libseccomp2@2.3.3-4, apt@1.8.2

  From: libseccomp/libseccomp2@2.3.3-4

  From: apt@1.8.2 > libseccomp/libseccomp2@2.3.3-4


✗ Low severity vulnerability found in libgcrypt20

  Description: CVE-2021-33560

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-1297893

  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2

  From: libgcrypt20@1.8.4-5

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5


✗ Low severity vulnerability found in libgcrypt20

  Description: Use of a Broken or Risky Cryptographic Algorithm

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-391902

  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2

  From: libgcrypt20@1.8.4-5

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5


✗ Low severity vulnerability found in gnutls28/libgnutls30

  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-340755

  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2

  From: gnutls28/libgnutls30@3.6.7-4

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4


✗ Low severity vulnerability found in gnupg2/gpgv

  Description: Use of a Broken or Risky Cryptographic Algorithm

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUPG2-535553

  Introduced through: gnupg2/gpgv@2.2.12-1+deb10u1, apt@1.8.2

  From: gnupg2/gpgv@2.2.12-1+deb10u1

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Double Free

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1078993

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Uncontrolled Recursion

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-338106

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Low severity vulnerability found in glibc/libc-bin

  Description: Uncontrolled Recursion

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-338163
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356371
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Resource Management Errors
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356671
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Resource Management Errors
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356735
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: CVE-2010-4051
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-356875
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Out-of-Bounds
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-452228
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Access Restriction Bypass
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-452267
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Use of Insufficiently Random Values
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-453375
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Information Exposure
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-453640
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Information Exposure
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-534995
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in glibc/libc-bin
  Description: Integer Underflow
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-564233
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ Low severity vulnerability found in coreutils
  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-COREUTILS-317465
  Introduced through: coreutils@8.30-3
  From: coreutils@8.30-3

✗ Low severity vulnerability found in coreutils
  Description: Race Condition
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-COREUTILS-317494
  Introduced through: coreutils@8.30-3
  From: coreutils@8.30-3

✗ Low severity vulnerability found in bash
```

```
  Description: Improper Check for Dropped Privileges
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-BASH-536280
  Introduced through: bash@5.0-4
  From: bash@5.0-4

✗ Low severity vulnerability found in apt/libapt-pkg5.0
  Description: Improper Verification of Cryptographic Signature
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-407502
  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2
  From: apt/libapt-pkg5.0@1.8.2
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2
  From: apt@1.8.2

✗ Medium severity vulnerability found in pcre3/libpcre3
  Description: Integer Overflow or Wraparound
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-572367
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12

✗ Medium severity vulnerability found in p11-kit/libp11-kit0
  Description: Out-of-bounds Read
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050832
  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2
  From: p11-kit/libp11-kit0@0.23.15-2
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2
  Fixed in: 0.23.15-2+deb10u1

✗ Medium severity vulnerability found in libzstd/libzstd1
```

```
  Description: Incorrect Default Permissions

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBZSTD-1080893

  Introduced through: libzstd/libzstd1@1.3.8+dfsg-3, apt@1.8.2

  From: libzstd/libzstd1@1.3.8+dfsg-3

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > libzstd/libzstd1@1.3.8+dfsg-3

  Fixed in: 1.3.8+dfsg-3+deb10u1


✗ Medium severity vulnerability found in libzstd/libzstd1

  Description: Incorrect Default Permissions

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBZSTD-1080899

  Introduced through: libzstd/libzstd1@1.3.8+dfsg-3, apt@1.8.2

  From: libzstd/libzstd1@1.3.8+dfsg-3

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > libzstd/libzstd1@1.3.8+dfsg-3

  Fixed in: 1.3.8+dfsg-3+deb10u2


✗ Medium severity vulnerability found in libgcrypt20

  Description: Race Condition

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-460489

  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2

  From: libgcrypt20@1.8.4-5

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Loop with Unreachable Exit Condition ('Infinite Loop')

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1035462

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Out-of-bounds Read

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1055403

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Out-of-Bounds

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559181

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in e2fsprogs/libcom-err2

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-E2FSPROGS-540890

  Introduced through: e2fsprogs/libcom-err2@1.44.5-1+deb10u2,
e2fsprogs@1.44.5-1+deb10u2, e2fsprogs/libext2fs2@1.44.5-1+deb10u2,
e2fsprogs/libss2@1.44.5-1+deb10u2

  From: e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  From: e2fsprogs@1.44.5-1+deb10u2 > e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  From: e2fsprogs@1.44.5-1+deb10u2 > e2fsprogs/libss2@1.44.5-1+deb10u2 >
e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  and 5 more...

  Fixed in: 1.44.5-1+deb10u3


✗ Medium severity vulnerability found in apt/libapt-pkg5.0
```

```
  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-1049974

  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2

  From: apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2

  Fixed in: 1.8.2.2


✗ Medium severity vulnerability found in apt/libapt-pkg5.0

  Description: Improper Input Validation

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-568926

  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2

  From: apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2

  Fixed in: 1.8.2.1


✗ High severity vulnerability found in systemd/libsystemd0

  Description: Privilege Chaining

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-345386

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ High severity vulnerability found in systemd/libsystemd0
```

```
  Description: Incorrect Privilege Assignment

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-345391

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ High severity vulnerability found in systemd/libsystemd0

  Description: Use After Free

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-546475

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...

  Fixed in: 241-7~deb10u4


✗ High severity vulnerability found in perl/perl-base

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570792

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in perl/perl-base
```

```
  Description: Buffer Overflow

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570797

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in perl/perl-base

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570802

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in p11-kit/libp11-kit0

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050833

  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2

  From: p11-kit/libp11-kit0@0.23.15-2

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2

  Fixed in: 0.23.15-2+deb10u1


✗ High severity vulnerability found in p11-kit/libp11-kit0

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050836

  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2

  From: p11-kit/libp11-kit0@0.23.15-2

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2

  Fixed in: 0.23.15-2+deb10u1
```

```
✗ High severity vulnerability found in nettle/libnettle6

  Description: Use of a Broken or Risky Cryptographic Algorithm

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-NETTLE-1090205

  Introduced through: nettle/libnettle6@3.4.1-1, apt@1.8.2,
nettle/libhogweed4@3.4.1-1

  From: nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libhogweed4@3.4.1-1
> nettle/libnettle6@3.4.1-1

  and 2 more...


✗ High severity vulnerability found in libidn2/libidn2-0

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBIDN2-474091

  Introduced through: libidn2/libidn2-0@2.0.5-1, apt@1.8.2

  From: libidn2/libidn2-0@2.0.5-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libidn2/libidn2-0@2.0.5-1

  Fixed in: 2.0.5-1+deb10u1


✗ High severity vulnerability found in libidn2/libidn2-0

  Description: Improper Input Validation

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBIDN2-474100

  Introduced through: libidn2/libidn2-0@2.0.5-1, apt@1.8.2

  From: libidn2/libidn2-0@2.0.5-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libidn2/libidn2-0@2.0.5-1


✗ High severity vulnerability found in gnutls28/libgnutls30

  Description: Use After Free

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-1085094
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-1085097
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use of a Broken or Risky Cryptographic Algorithm
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-564383
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4
  Fixed in: 3.6.7-4+deb10u3

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use of a Broken or Risky Cryptographic Algorithm
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-571282
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4
  Fixed in: 3.6.7-4+deb10u4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Out-of-bounds Write
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-609778
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in glibc/libc-bin
  Description: Reachable Assertion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1065768
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in glibc/libc-bin
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1296899
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in glibc/libc-bin
  Description: Out-of-bounds Write
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559488
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in glibc/libc-bin
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559493
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in gcc-8/libstdc++6
  Description: Information Exposure
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GCC8-347558
  Introduced through: gcc-8/libstdc++6@8.3.0-6, apt@1.8.2,
meta-common-packages@meta
  From: gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  and 2 more...

✗ High severity vulnerability found in gcc-8/libstdc++6
  Description: Insufficient Entropy
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GCC8-469413
  Introduced through: gcc-8/libstdc++6@8.3.0-6, apt@1.8.2,
meta-common-packages@meta
  From: gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  and 2 more...



Organization:      dylanlott
Package manager:   deb
Project name:      docker-image|ethersphere/clef
Docker image:      ethersphere/clef:0
```

```
Platform:          linux/amd64

Licenses:          enabled


Tested 85 dependencies for known issues, found 76 issues.y vulnerability found
in coreutils

  Description: Race Condition

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-COREUTILS-317494

  Introduced through: coreutils@8.30-3

  From: coreutils@8.30-3


✗ Low severity vulnerability found in bash

  Description: Improper Check for Dropped Privileges

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-BASH-536280

  Introduced through: bash@5.0-4

  From: bash@5.0-4


✗ Low severity vulnerability found in apt/libapt-pkg5.0

  Description: Improper Verification of Cryptographic Signature

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-407502

  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2

  From: apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2


✗ Medium severity vulnerability found in pcre3/libpcre3

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PCRE3-572367

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > pcre3/libpcre3@2:8.39-12
```

```
✗ Medium severity vulnerability found in p11-kit/libp11-kit0
  Description: Out-of-bounds Read
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050832
  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2
  From: p11-kit/libp11-kit0@0.23.15-2
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2
  Fixed in: 0.23.15-2+deb10u1

✗ Medium severity vulnerability found in libzstd/libzstd1
  Description: Incorrect Default Permissions
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBZSTD-1080893
  Introduced through: libzstd/libzstd1@1.3.8+dfsg-3, apt@1.8.2
  From: libzstd/libzstd1@1.3.8+dfsg-3
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > libzstd/libzstd1@1.3.8+dfsg-3
  Fixed in: 1.3.8+dfsg-3+deb10u1

✗ Medium severity vulnerability found in libzstd/libzstd1
  Description: Incorrect Default Permissions
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBZSTD-1080899
  Introduced through: libzstd/libzstd1@1.3.8+dfsg-3, apt@1.8.2
  From: libzstd/libzstd1@1.3.8+dfsg-3
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > libzstd/libzstd1@1.3.8+dfsg-3
  Fixed in: 1.3.8+dfsg-3+deb10u2

✗ Medium severity vulnerability found in libgcrypt20
  Description: Race Condition
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBGCRYPT20-460489
  Introduced through: libgcrypt20@1.8.4-5, apt@1.8.2

  From: libgcrypt20@1.8.4-5

  From: apt@1.8.2 > gnupg2/gpgv@2.2.12-1+deb10u1 > libgcrypt20@1.8.4-5

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2 > libgcrypt20@1.8.4-5


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Loop with Unreachable Exit Condition ('Infinite Loop')

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1035462

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Out-of-bounds Read

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1055403

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in glibc/libc-bin

  Description: Out-of-Bounds

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559181

  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta

  From: glibc/libc-bin@2.28-10

  From: meta-common-packages@meta > glibc/libc6@2.28-10


✗ Medium severity vulnerability found in e2fsprogs/libcom-err2

  Description: Out-of-bounds Write
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-E2FSPROGS-540890

  Introduced through: e2fsprogs/libcom-err2@1.44.5-1+deb10u2,
e2fsprogs@1.44.5-1+deb10u2, e2fsprogs/libext2fs2@1.44.5-1+deb10u2,
e2fsprogs/libss2@1.44.5-1+deb10u2

  From: e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  From: e2fsprogs@1.44.5-1+deb10u2 > e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  From: e2fsprogs@1.44.5-1+deb10u2 > e2fsprogs/libss2@1.44.5-1+deb10u2 >
e2fsprogs/libcom-err2@1.44.5-1+deb10u2

  and 5 more...

  Fixed in: 1.44.5-1+deb10u3


✗ Medium severity vulnerability found in apt/libapt-pkg5.0

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-1049974

  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2

  From: apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2

  Fixed in: 1.8.2.2


✗ Medium severity vulnerability found in apt/libapt-pkg5.0

  Description: Improper Input Validation

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-APT-568926

  Introduced through: apt/libapt-pkg5.0@1.8.2, apt@1.8.2

  From: apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2

  From: apt@1.8.2

  Fixed in: 1.8.2.1


✗ High severity vulnerability found in systemd/libsystemd0
```

```
  Description: Privilege Chaining

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-345386

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ High severity vulnerability found in systemd/libsystemd0

  Description: Incorrect Privilege Assignment

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-345391

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2

  and 4 more...


✗ High severity vulnerability found in systemd/libsystemd0

  Description: Use After Free

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-SYSTEMD-546475

  Introduced through: systemd/libsystemd0@241-7~deb10u2,
util-linux/bsdutils@1:2.33.1-0.1, apt@1.8.2, util-linux/mount@2.33.1-0.1,
systemd/libudev1@241-7~deb10u2

  From: systemd/libsystemd0@241-7~deb10u2

  From: util-linux/bsdutils@1:2.33.1-0.1 > systemd/libsystemd0@241-7~deb10u2

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 >
systemd/libsystemd0@241-7~deb10u2
```

```
  and 4 more...

  Fixed in: 241-7~deb10u4


✗ High severity vulnerability found in perl/perl-base

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570792

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in perl/perl-base

  Description: Buffer Overflow

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570797

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in perl/perl-base

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570802

  Introduced through: meta-common-packages@meta

  From: meta-common-packages@meta > perl/perl-base@5.28.1-6

  Fixed in: 5.28.1-6+deb10u1


✗ High severity vulnerability found in p11-kit/libp11-kit0

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050833

  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2

  From: p11-kit/libp11-kit0@0.23.15-2
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2

  Fixed in: 0.23.15-2+deb10u1


✗ High severity vulnerability found in p11-kit/libp11-kit0

  Description: Integer Overflow or Wraparound

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-P11KIT-1050836

  Introduced through: p11-kit/libp11-kit0@0.23.15-2, apt@1.8.2

  From: p11-kit/libp11-kit0@0.23.15-2

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 >
p11-kit/libp11-kit0@0.23.15-2

  Fixed in: 0.23.15-2+deb10u1


✗ High severity vulnerability found in nettle/libnettle6

  Description: Use of a Broken or Risky Cryptographic Algorithm

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-NETTLE-1090205

  Introduced through: nettle/libnettle6@3.4.1-1, apt@1.8.2,
nettle/libhogweed4@3.4.1-1

  From: nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libnettle6@3.4.1-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > nettle/libhogweed4@3.4.1-1
> nettle/libnettle6@3.4.1-1

  and 2 more...


✗ High severity vulnerability found in libidn2/libidn2-0

  Description: Out-of-bounds Write

  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBIDN2-474091

  Introduced through: libidn2/libidn2-0@2.0.5-1, apt@1.8.2

  From: libidn2/libidn2-0@2.0.5-1

  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libidn2/libidn2-0@2.0.5-1

  Fixed in: 2.0.5-1+deb10u1
```

```
✗ High severity vulnerability found in libidn2/libidn2-0
  Description: Improper Input Validation
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-LIBIDN2-474100
  Introduced through: libidn2/libidn2-0@2.0.5-1, apt@1.8.2
  From: libidn2/libidn2-0@2.0.5-1
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4 > libidn2/libidn2-0@2.0.5-1

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-1085094
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-1085097
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use of a Broken or Risky Cryptographic Algorithm
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-564383
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4
  Fixed in: 3.6.7-4+deb10u3
```

```
✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Use of a Broken or Risky Cryptographic Algorithm
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-571282
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4
  Fixed in: 3.6.7-4+deb10u4

✗ High severity vulnerability found in gnutls28/libgnutls30
  Description: Out-of-bounds Write
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GNUTLS28-609778
  Introduced through: gnutls28/libgnutls30@3.6.7-4, apt@1.8.2
  From: gnutls28/libgnutls30@3.6.7-4
  From: apt@1.8.2 > gnutls28/libgnutls30@3.6.7-4

✗ High severity vulnerability found in glibc/libc-bin
  Description: Reachable Assertion
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1065768
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in glibc/libc-bin
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-1296899
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10
```

*This audit makes no statements or warranties and is for discussion purposes only.*

```
✗ High severity vulnerability found in glibc/libc-bin
  Description: Out-of-bounds Write
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559488
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in glibc/libc-bin
  Description: Use After Free
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GLIBC-559493
  Introduced through: glibc/libc-bin@2.28-10, meta-common-packages@meta
  From: glibc/libc-bin@2.28-10
  From: meta-common-packages@meta > glibc/libc6@2.28-10

✗ High severity vulnerability found in gcc-8/libstdc++6
  Description: Information Exposure
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GCC8-347558
  Introduced through: gcc-8/libstdc++6@8.3.0-6, apt@1.8.2,
meta-common-packages@meta
  From: gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > gcc-8/libstdc++6@8.3.0-6
  and 2 more...

✗ High severity vulnerability found in gcc-8/libstdc++6
  Description: Insufficient Entropy
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-GCC8-469413
  Introduced through: gcc-8/libstdc++6@8.3.0-6, apt@1.8.2,
meta-common-packages@meta
```

```
  From: gcc-8/libstdc++6@8.3.0-6

  From: apt@1.8.2 > gcc-8/libstdc++6@8.3.0-6

  From: apt@1.8.2 > apt/libapt-pkg5.0@1.8.2 > gcc-8/libstdc++6@8.3.0-6

  and 2 more...

Project name:      docker-image|ethersphere/clef

Docker image:      ethersphere/clef:0

Platform:          linux/amd64

Licenses:          enabled


Tested 85 dependencies for known issues, found 76 issues.
```

Alpine 3.9.4 is no longer supported by the Alpine maintainers. Vulnerability detection may be affected by a lack of security updates.

# About Least Authority

We believe that people have a fundamental right to privacy and that the use of secure solutions enables people to more freely use the Internet and other connected technologies. We provide security consulting services to help others make their solutions more resistant to unauthorized access to data and unintended manipulation of the system. We support teams from the design phase through the production launch and after.

The Least Authority team has skills for reviewing code in C, C++, Python, Haskell, Rust, Node.js, Solidity, Go, and JavaScript for common security vulnerabilities and specific attack vectors. The team has reviewed implementations of cryptographic protocols and distributed system architecture, including in cryptocurrency, blockchains, payments, and smart contracts. Additionally, the team can utilize various tools to scan code and networks and build custom tools as necessary.

Least Authority was formed in 2011 to create and further empower freedom-compatible technologies. We moved the company to Berlin in 2016 and continue to expand our efforts. Although we are a small team, we believe that we can have a significant impact on the world by being transparent and open about the work we do.

For more information about our security consulting, please visit https://leastauthority.com/security-consulting/.

# Our Methodology

We like to work with a transparent process and make our reviews a collaborative effort. The goals of our security audits are to improve the quality of systems we review and aim for sufficient remediation to help protect users. The following is the methodology we use in our security audit process.

## Manual Code Review

In manually reviewing all of the code, we look for any potential issues with code logic, error handling, protocol and header parsing, cryptographic errors, and random number generators. We also watch for areas where more defensive programming could reduce the risk of future mistakes and speed up future audits. Although our primary focus is on the in-scope code, we examine dependency code and behavior when it is relevant to a particular line of investigation.

## Vulnerability Analysis

Our audit techniques included manual code analysis, user interface interaction, and whitebox penetration testing. We look at the project's web site to get a high level understanding of what functionality the software under review provides. We then meet with the developers to gain an appreciation of their vision of the software. We install and use the relevant software, exploring the user interactions and roles. While we do this, we brainstorm threat models and attack surfaces. We read design documentation, review other audit results, search for similar projects, examine source code dependencies, skim open issue tickets, and generally investigate details other than the implementation. We hypothesize what vulnerabilities may be present, creating Issue entries, and for each we follow the following Issue Investigation and Remediation process.

*This audit makes no statements or warranties and is for discussion purposes only.*

## Documenting Results

We follow a conservative, transparent process for analyzing potential security vulnerabilities and seeing them through successful remediation. Whenever a potential issue is discovered, we immediately create an Issue entry for it in this document, even though we have not yet verified the feasibility and impact of the issue. This process is conservative because we document our suspicions early even if they are later shown to not represent exploitable vulnerabilities. We generally follow a process of first documenting the suspicion with unresolved questions, then confirming the issue through code analysis, live experimentation, or automated tests. Code analysis is the most tentative, and we strive to provide test code, log captures, or screenshots demonstrating our confirmation. After this we analyze the feasibility of an attack in a live system.

## Suggested Solutions

We search for immediate mitigations that live deployments can take, and finally we suggest the requirements for remediation engineering for future releases. The mitigation and remediation recommendations should be scrutinized by the developers and deployment engineers, and successful mitigation and remediation is an ongoing collaborative process after we deliver our report, and before the details are made public.

## Responsible Disclosure

Before our report or any details about our findings and suggested solutions are made public, we like to work with your team to find reasonable outcomes that can be addressed as soon as possible without an overly negative impact on pre-existing plans. Although the handling of issues must be done on a case-by-case basis, we always like to agree on a timeline for resolution that balances the impact on the users and the needs of your project team. We take this agreed timeline into account before publishing any reports to avoid the necessity for full disclosure.