Introducing the SecureDrop Protocol

IETF 120
Giulio Berra • Freedom of the Press Foundation
SecureDrop today: architecture

OpenPGP encryption at rest

Tor encryption in transit
E2EE SecureDrop: goals

- Use modern end-to-end encryption
- Do not require file or information persistence on the source/whistleblower side
- Ensure system architecture does not preclude deployment in a hostile, potentially compromised environment
- Avoid, minimize, and hide metadata from the server
E2EE SecureDrop: non-goals

- High volume, high traffic
- Low latency
- Arbitrary-sized groups
- Arbitrary direct messages
- Federation
E2EE SecureDrop: properties

1. No accounts, and therefore no user authentication
2. No message flow metadata
3. No changes in server state are observable externally
4. No ciphertext collection or information leaks
Comparison of E2EE messaging protocols

<table>
<thead>
<tr>
<th>Approach</th>
<th>Primitives Library</th>
<th>Peer-reviewed Library</th>
<th>Groups</th>
<th>Scalable</th>
<th>Concealed Recipient</th>
<th>Private Server State</th>
<th>Avoids Accounts</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signal</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Trial Decryption</td>
<td>Yes</td>
<td>N/A¹</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Oblivious Message Retrieval</td>
<td>No</td>
<td>No</td>
<td>Yes²</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>SecureDrop Protocol</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No³</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

1. While there isn’t a single standard library, the implementation is straightforward.
2. **New iteration of the research focuses on groups.**
3. SecureDrop Protocol does not preclude scalability, but scaling to mass adoption level (i.e. millions of users) is a **nonrequirement** for our purposes.
E2EE SecureDrop: high-level flow

Sender → Payload, Clue → SD Server → Remixed Clues → Recipient

SD Server:
- Payload ID
- Payload

IDs, Clues, Payloads

Trial decryption on all remixed clues to discover payload ID, payload decryption
Most whistleblowing software/services have a similar setup: web-based and single server

Could a stable, maintained library improve the ecosystem? Even the commercial ones?

Is the threat model accurate, realistic, and broad enough?

Is the protocol portable in a PQ world? (3-party commutative DH is not trivial)

What other countermeasures are needed? Decoy/noise traffic?
Acknowledgments and ongoing work

- **Preliminary cryptographic audit** done by Michele Orrù (French National Center for Cryptographic Research/CNRS)
- **Funded** by the Filecoin Foundation for the Decentralized Web
- **Formal analysis** in progress by Luca Maier supervised by Felix Linker (Swiss Federal Institute of Technology/ETH Zürich)
Our ask: your feedback

Read more:

- https://github.com/freedomofpress/securedrop-protocol

Write to us:

- giulio@freedom.press (or Signal: giulio.99)
- cory@freedom.press (or Signal: cfm.38)