Testing & Evaluation Sheet

Magic Wormhole

1. Tool Overview

Name:	Magic Wormhole
Category:	File Transfer
Purpose:	Provides a library & a command line so users can send arbitrary sized files and directories from one computer to another
Date Tested	4/2/25
Status:	Deployed ☑ Operational - Actively running/maintained □ In Testing - Currently being evaluated or piloted □ Inactive/Deprecated - No longer maintained or functional
Deployment Architecture:	□ A standalone software - Runs entirely locally (e.g., runs on computer and doesn't depend on external server) □ A locally hosted service with separate server and client component - Run both backend/frontend yourself (e.g., backend could be on a local network, or self-hosted on cloud) ☑ A service with a local client that's hosted by a third party - You install a client on your device, but it connects to and depends on a remote server (e.g., Signal: install app (client), but Signal's servers handle message relaying, etc.) □ A service that is hosted by a third party but can also be self-hosted
Version:	V 0.6.3

2. Installation & Setup

OS Compatibility	MacOS, Linux, Windows
Installation Manual:	Yes
Installation Steps:	1. Open Terminal as Administrator: a. Windows: Right-click **Start** > "**Command Prompt (Admin)**", "**Windows PowerShell (Admin)**", or "**Terminal (Admin)**".

	b. macOS/Linux: Open **Terminal**. 2. Install Wormhole: a. Follow installation instructions for your OS in the documentation above. 3. Accept the Download: a. Type "**y**" or "**a**" when prompted to accept the download.
Mention if command-line setup or special configurations are needed	Entire tool is a command-line tool which can be intimidating but does not require special configurations. It is relatively easy to use despite using the command line as the interface for file transfer.
Common Installation Issues & Fixes:	● A common issue users face with Magic Wormhole is locating the downloaded file after transfer. By default, the file is saved in the current working directory of the terminal. Users need to be aware of their system's file path and directory structure to know where downloads are saved ● A common issue for users sharing a file is knowing how to correctly copy the file path into the terminal. To send a file, the full file path must be entered after the wormhole send command.
User Documentation:	Yes
Required Technical Knowledge	Intermediate

3. Testing & Evaluation

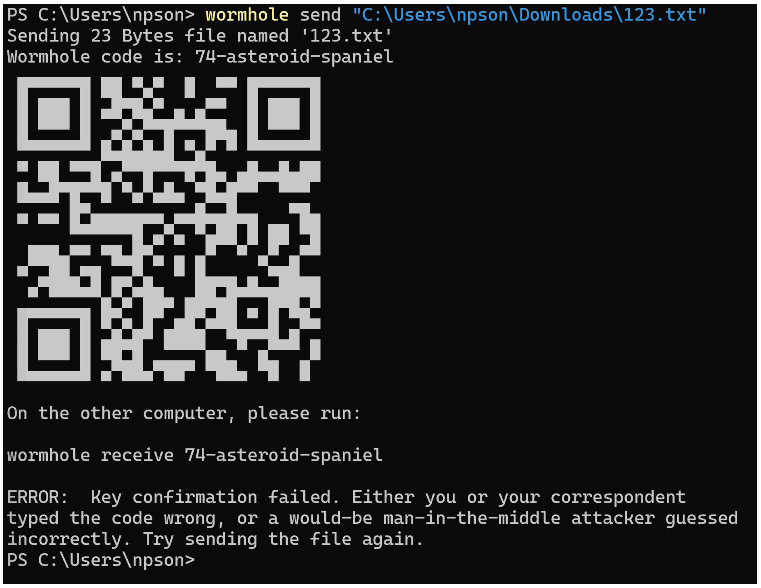
<u>Category</u>	<u>Details</u>	<u>Score</u>
Operational Functionality:	**Functionality** ● Magic Wormhole effectively enables secure file transfers between devices while implementing strong security measures. It utilizes a structured protocol involving a Mailbox Server, Transit Relay, and Dilation Protocol to facilitate encrypted, peer-to-peer communication. The system ensures reliable data transmission even in cases of network interruptions. ● No broken features noticed ☐ The tool is mostly non-functional with many broken features and bugs.	3.3

	□ Several broken features or bugs □ Minor bugs or issues □ Mostly functional with few bugs or no bugs ☑ Fully functional with no bugs **Internet Dependence:** ● Does not have offline functionality, must connect to relay servers **Localization & Language Support** ● Only English available ● Community does not seem to to be working on language localization **Mobile Accessibility** ● Not available on mobile devices, needs a computer in order to use the terminal to send the files.	
Usability for Non-Technical Users	**Ease of Installation & Deployment** ● 4 steps are required ● Requires the use of command lines ● Well-maintained setup guides and FAQs ● Has extensive for installation, usage, and security information ● Installation takes <2 minutes ● Figuring out the available tags/functions wormhole supports such as Tor may be harder for new users to find. **User Onboarding Experience** ● Has extensive documentation ranging from installation, implementations & support, tor support, etc. ● https://magic-wormhole.readthedocs.io/en/latest/ **Technical Experience Level Required** ● Yes the only intimidating part is navigating the terminal but it isn't that difficult. ● Relies heavily on command lines in terminal	4.3
Security & Privacy Strength	**Encryption Standards** ● Wormhole codes contain 16 bits of entropy making brute-force guessing highly unlikely (1 in 65,536 chance). ● Could be blocked if authorities control the network or the mailbox server. **Censorship resilience** ● Can be usable in regions with heavy censorship or surveillance when configured with Tor	4.6

	● Does not include built-in circumvention tools **Vulnerability: Its Resilience against known threats** ● Man-in-the-Middle (MitM) attacks are possible if an attacker intercepts traffic and repeatedly guesses codes. Longer codes (--code-length=4) mitigate this risk. ● Magic Wormhole's rendezvous server is a single point of failure (SPOF) vulnerable to DoS attacks, where an attacker can brute-force nameplates to disrupt key exchanges, but the protocol includes a "permission" feature allowing proof-of-work challenges (e.g., HashCash) to mitigate such attacks. **Comparison with Known Standards** ● Compared to stronger systems like TLS or PGP, the use of a 16-bit code in Magic Wormhole is slightly insecure. ● The encryption used in Magic Wormhole (NaCl "secretbox") is strong and reliable for small, fast communication **Data Minimization** ● Only the necessary data (file transfer metadata) is processed. **Privacy Policy Accessibility and Clarity** ● The privacy policy is clear about data handling and provides considerations to be more secure and private. ● https://github.com/magic-wormhole/magic-wormhole-protocols/security/policy	
Maintenance/Sustainability	**Community support** ● The community is active and there are regular updates. ● It is easy to get help and ask questions, or find solutions from developers **Development active status** ● Updated at least once a month ● Last updated December 2024 ● The development team is responsive to good changes. **Funding and Sponsorship** ● No clear government funding ● *Appears to be done by an individual , which may support neutrality* ● Overall is financially stable	3.0

Performance / Effectiveness & Reliability	**Testing Environment Setup:** ● **Device:** HP Envy x360 ○ 13th Gen Intel(R) i7 processor ○ 16 GB RAM ● **Windows 11** ● **Network:** 4G Network **User Experience Observations** ● Minor load time for sending files ● Slight delay of response when using the computer terminal to send files **Speed & Responsiveness:** ● Near-instant setup and initialization ● Transfer starts immediately after sender and receiver enter passcode/command. ● Are there any noticeable delays or lag during use? **Resource Usage:** ● Minimal for small file transfers but increases with large files (1-5% CPU usage). ● Small files: 10-50MB RAM and large files can be 200MB+ RAM. **Network Performance:** ● Uses full available bandwidth if sending over direct peer-to-peer (P2P). If using a relay server, speeds may slow depending on congestion. ● Latency: roughly 3-20 ms for smaller file size, 10-50 ms for medium file size, and100ms+ for larger file size depending on relay server use (Tor). ● Bandwidth usage is max available for peer-to-peer connections but limited by relay (if using Tor) **Reliability** ● Many trust Magic Wormhole to be secure and reliable along with an extensive team of developers on the Github that assist in improving the tool.	4.5
Deployment Considerations:	**Open Source & Transparency:** ● Yes the code is open for independent verification on the Github **Cloud vs. Local Deployment:** ● Can be run locally without requiring AWS/Azure. **Dependencies:** ● Requires Python but does not rely on Docker or databases ● Dependencies are clearly documented **Post-Deployment Maintenance** ● Easy to maintain after deployment. ● Yes it is easy to modify the UI but cryptographic algorithms may require higher expertise.	

	Merge/Sustainability: ● The project is open to contributions ● Submitting changes to the main repository is relatively easy if there are good changes.
4. Testing Scenarios	
How To Use (Basic)	**Send a File:** ● To send a file to another computer type: **wormhole send [filename/filepath]** ○ Ex: wormhole send “C:\Users\person\abc.txt” ● A “magic-code” will be generated upon sending which the receiver will need. **Receive a File:** ○ To receive a file from another computer type: **wormhole receive [magic-code]** ○ The “magic-code” would be provided by the sender. ○ By default, wormhole receive [magic-code] saves the file in the current directory ■ To save the file to a specific directory or rename it, use tag: **--output-file [filename/filepath]** ■ Rename: ● Ex: wormhole receive 7-chicken-monster --output-file "C:\Users\person\Downloads\received_file.txt” ● Ensure that the received file retains its original file type if renaming (e.g., if the file is a .png, it should remain a .png). ■ Choose Directory: ● Ex: wormhole receive 23-purple-dragon --output-file "C:\Users\person\Downloads\”

- 

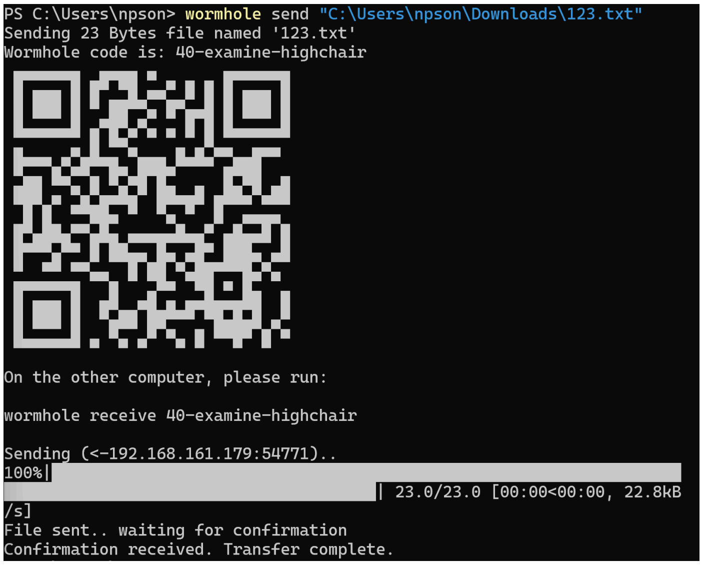
  ```
  PS C:\Users\npson> wormhole send "C:\Users\npson\Downloads\123.txt"
  Sending 23 Bytes file named '123.txt'
  Wormhole code is: 74-asteroid-spaniel

  On the other computer, please run:

  wormhole receive 74-asteroid-spaniel

  ERROR:  Key confirmation failed. Either you or your correspondent
  typed the code wrong, or a would-be man-in-the-middle attacker guessed
  incorrectly. Try sending the file again.
  PS C:\Users\npson>
  ```

Figure 1: Writing the wrong "magic-code" results in this which shows the security when it comes to incorrectly typing the code and protecting against attackers.

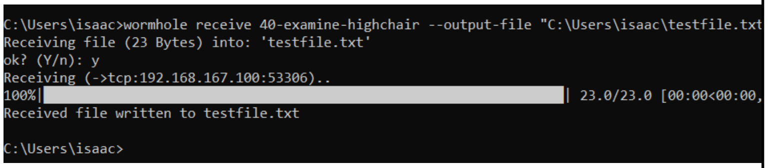
- 

  ```
  PS C:\Users\npson> wormhole send "C:\Users\npson\Downloads\123.txt"
  Sending 23 Bytes file named '123.txt'
  Wormhole code is: 40-examine-highchair

  On the other computer, please run:

  wormhole receive 40-examine-highchair

  Sending (<-192.168.161.179:54771)..
  100%|
                                       | 23.0/23.0 [00:00<00:00, 22.8kB
  /s]
  File sent.. waiting for confirmation
  Confirmation received. Transfer complete.
  ```

- 

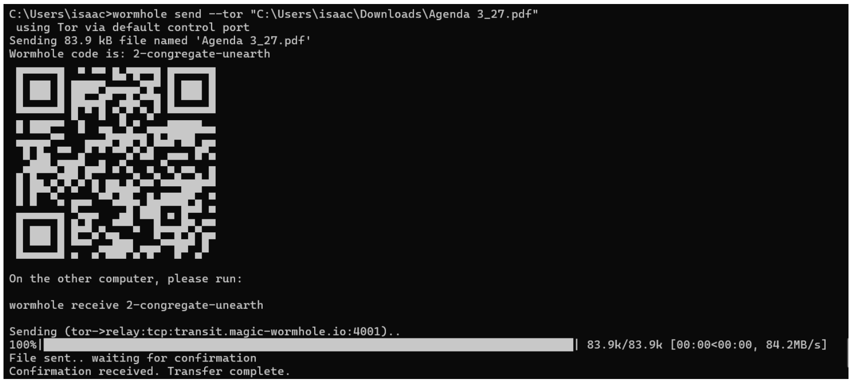
  ```
  C:\Users\isaac>wormhole receive 40-examine-highchair --output-file "C:\Users\isaac\testfile.txt
  Receiving file (23 Bytes) into: 'testfile.txt'
  ok? (Y/n): y
  Receiving (->tcp:192.168.167.100:53306)..
  100%|                                                     | 23.0/23.0 [00:00<00:00,
  Received file written to testfile.txt

  C:\Users\isaac>
  ```

Figure 2: The two images above are from two different laptops showing how the file was sent to the other laptop.

Using Magic Wormhole through Tor Relay Network

- Use the **--tor** flag in the command line, along with the parameters shown in Figures 3 and 4, to route traffic through Tor’s relay network. This hides the user's IP address so they can be anonymous.
    1. To utilize this feature, Tor must be running.



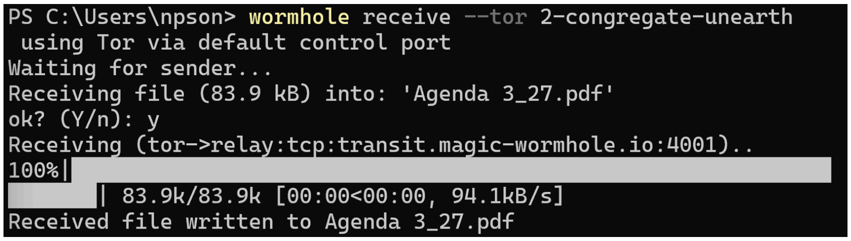
```
C:\Users\isaac>wormhole send --tor "C:\Users\isaac\Downloads\Agenda 3_27.pdf"
 using Tor via default control port
Sending 83.9 kB file named 'Agenda 3_27.pdf'
Wormhole code is: 2-congregate-unearth

On the other computer, please run:

wormhole receive 2-congregate-unearth

Sending (tor->relay:tcp:transit.magic-wormhole.io:4001)..
100%|████████████████████████████████████████████████| 83.9k/83.9k [00:00<00:00, 84.2MB/s]
File sent.. waiting for confirmation
Confirmation received. Transfer complete.
```

Figure 3: Sending file through wormhole using Tor’s relay network to be anonymous.



```
PS C:\Users\npson> wormhole receive --tor 2-congregate-unearth
 using Tor via default control port
Waiting for sender...
Receiving file (83.9 kB) into: 'Agenda 3_27.pdf'
ok? (Y/n): y
Receiving (tor->relay:tcp:transit.magic-wormhole.io:4001)..
100%|██████████████████████████████████████████████
█████| 83.9k/83.9k [00:00<00:00, 94.1kB/s]
Received file written to Agenda 3_27.pdf
```

Figure 4: Receiving file through wormhole using Tor’s relay network to be anonymous.

5. Insights & Recommendations

Key Findings

Strengths:

- Easy to use with a simple command-line interface for file and text transfers.
- Uses PAKE (Password-Authenticated Key Exchange) for secure key negotiation. Messages are encrypted, and the server cannot read contents.
- Works across different operating systems
- Supports text, file, and directory transfers, with API and library support for integration into other applications.
- Ability to use Tor relay network to transfer file anonymously

Weaknesses:

- IP Address has the potential to be leaked while transferring files, but can be combated by using Magic Wormhole through Tor.

	● The default relay server has no uptime guarantees, which can cause connection issues. ● The default 16-bit entropy in wormhole codes makes attacks possible (1 in 65,536 chance) ● The default relay server has no uptime guarantees, which can cause connection issues. ● Difficult to self-host/deploy.
Suggested Improvements	● Create quick video tutorials and interactive documentation for technical and non-technical users. ● By increasing the 16-bit PAKE code to 32 bits or more, the number of possible codes increases exponentially making brute-force attacks significantly harder. ● Magic Wormhole currently uses a fixed dictionary of words for human readability. Expanding the dictionary or using longer phrases would increase security while maintaining usability. ● Sending an additional verification step (like an email confirmation or a secondary secret key) can further strengthen security. ● While secretbox is secure for small messages, integrating additional cryptographic handshakes like TLS (for transport security) or leveraging Signal's Double Ratchet Algorithm for forward secrecy could improve security further.
Alternative Tools:	● Croc ● Send
License	GNU AGPL V3
Cost/Resource Implications	**Total Cost of Ownership:** ● Magic Wormhole is completely free to use
Why is this useful to civil societies in authoritarian environments?	● **Cross Platform and Peer-to-Peer:** An individual from one NGO can send a file to another NGO in a different country easily with peer-to-peer transfer. Also supports multiple platforms, making it useful across a variety of devices and operating systems. ● **Privacy:** Magic wormhole does not leak any Metadata which is ideal for whistleblowers or individuals sharing data in repressive environments ● **Secure File Transfer:** Magic-wormhole has End to End Encryption making it useful for sharing documents, media, or other reports with journalists or legal teams..

	● **Bypass Censorship:** Magic wormhole doesn’t rely on centralized infrastructure and can be more resilient in restricted networks. ● **Avoid Surveillance:** Files are encrypted end-to-end and transferred directly between peers