PTP 內容分享網路

PTP CONTENTS SHARING NETWORK

91d906h4

2022/07/24

摘要

長期以來，Bittorrent 一直無法解決匿名性的問題，Tracker 使得檔案下載者可能因 IP 公開而暴露於風險之中。這使得對匿名性要求高的使用者僅有較低的意願透過 Bittorrent 來下載及上傳檔案。而 Tor（The Onion Router）是一個匿名性極高的網路，透過伺服器與客戶端之間多個節點的保護，使雙方處在極度安全的網路環境。

而要達成高匿名性之大型檔案分享，現行的方法還是必須依賴主從式架構，而其中，對第三方的信任變成了必要條件。在 Web 2.0 的時代，人們似乎已習慣對第三方服務的信任與依賴，這使得他們變得愈來愈具權威，直到壟斷整個市場。

而 PTP（Peer over Tor to Peer）系統的目的正是解決這些問題。透過使用 P2P 及 Tor 相關技術，PTP 實現了高匿名性、高分散性、高安全性之檔案分享系統。

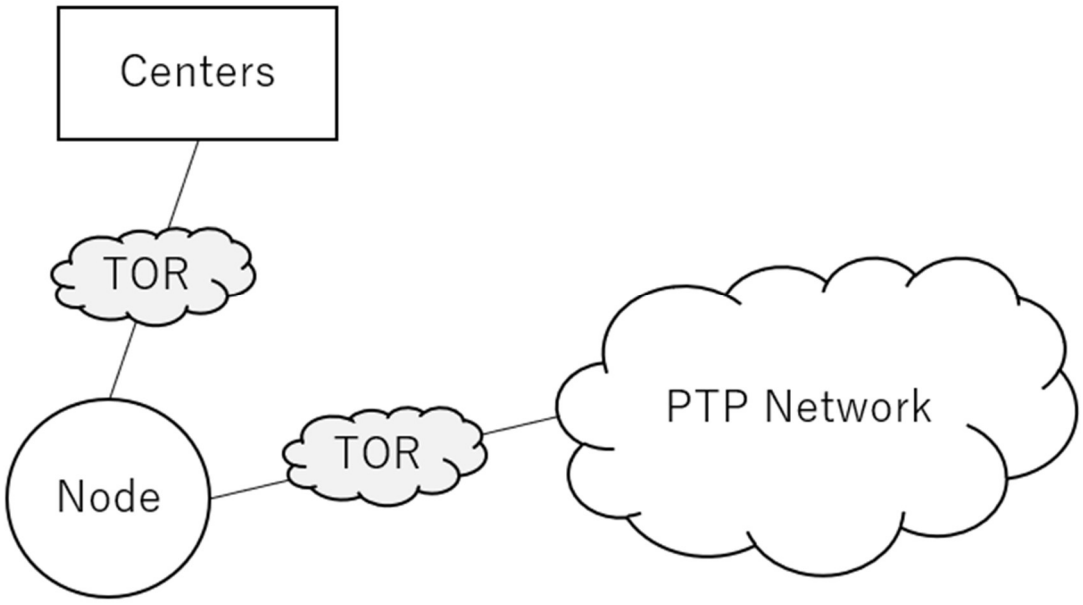
目錄

1. 用語

Address：識別節點之位址，即該節點之 .onion 位址。
Center：用以追蹤檔案分布的特殊節點。
Connection Code：PTP 協定中的連線代碼。格式為「PT-XXXX」。
Download Node：下載節點。表示在一次的檔案下在作業中，下載檔案得節點。
Node：一般節點。負責上傳、下載，及儲存檔案。
PTP Protocol：PTP 傳輸協定。
Seed：種子檔。提供檔案 splits 目錄以及 Center 目錄。下載時必須使用此檔案。
Source Node：來源節點。表示在一次的檔案下在作業中，提供檔案得節點。
Split(s)：分割檔。大型檔案分割後產生的檔案。副檔名為 .ptp。

2. PTP 系統架構

在 PTP 系統中，節點主要可分為兩種，Center 及 Node。雖然 PTP 是對等網路，但為了提升效率，PTP 在節點之中加入了名為 Center（用於追蹤檔案）的特殊節點。該節點不儲存任何使用者檔案，而是儲存檔案當前的分布位置。這將協助 Node（檔案下載及上傳者）找到檔案目前分布於哪些節點之中。而下面的示意圖說明了單一節點與 PTP 網路的關係：



PTP 節點與網路

所有的連線都將透過 Tor 進行，而節點之間的識別則完全透過 .onion 位址實現，這將確保所有節點都無法得知對方的 IP，進而達成匿名的目的。

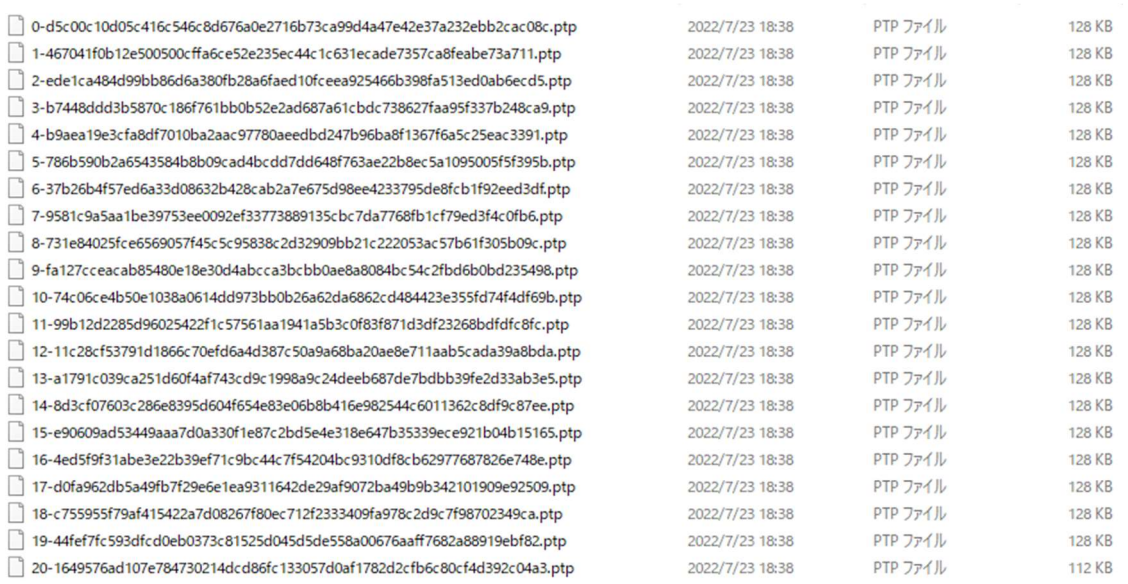
PTP 系統主要以 PHP 編寫。

3. PTP 運作原理

3.1 檔案的分割

節點將一個檔案公開至 PTP 網路前，必須先將檔案分割。分割檔案的主要原因有幾個：1. 避免檔案傳送錯誤時，必須重新傳送整個檔案。透過網路傳輸，無可避免的，在一定的機率下會產生錯誤，這時，下載節點（欲下載檔案的節點）必須重新向來源方要求檔案，若將檔案加以分割並傳送，則只需要求出現錯誤的 splits 即可；2. 避免節點頻寬被長時間占用。當同一個 Node 向同樣的另一個 Node 傳送過多下載要求，則檔案所在的 Node 將拒絕更多的下載要求。

分割檔案的作業將在本地（local）進行，並且以每個 split 128KB 為上限進行分割。分割完成後的 splits 將被保留在本地，並且以「序號-SHA256 雜湊值」的格式命名，而檔案目錄及原始 Node 的位址將被上傳至 Node 指定之 Centers。以下為某個檔案分割後所產生之 splits：



0-d5c00c10d05c416c546c8d676a0e2716b73ca99d4a47e42e37a232ebb2cac08c.ptp	2022/7/23 18:38	PTP ファイル	128 KB
1-467041f0b12e500500cffa6ce52e235ec44c1c631ecade7357ca8feabe73a711.ptp	2022/7/23 18:38	PTP ファイル	128 KB
2-ede1ca484d99bb86d6a380fb28a6faed10fceea925466b398fa513ed0ab6ecd5.ptp	2022/7/23 18:38	PTP ファイル	128 KB
3-b7448ddd3b5870c186f761bb0b52e2ad687a61cbdc738627faa95f337b248ca9.ptp	2022/7/23 18:38	PTP ファイル	128 KB
4-b9aea19e3cfa8df7010ba2aac97780aeedbd247b96ba8f1367f6a5c25eac3391.ptp	2022/7/23 18:38	PTP ファイル	128 KB
5-786b590b2a6543584b8b09cad4bcdd7dd648f763ae22b8ec5a1095005f5f395b.ptp	2022/7/23 18:38	PTP ファイル	128 KB
6-37b26b4f57ed6a33d08632b428cab2a7e675d98ee4233795de8fcb1f92eed3df.ptp	2022/7/23 18:38	PTP ファイル	128 KB
7-9581c9a5aa1be39753ee0092ef33773889135cbc7da7768fb1cf79ed3f4c0fb6.ptp	2022/7/23 18:38	PTP ファイル	128 KB
8-731e84025fce6569057f45c5c95838c2d32909bb21c222053ac57b61f305b09c.ptp	2022/7/23 18:38	PTP ファイル	128 KB
9-fa127cceacab85480e18e30d4abcca3bcbb0ae8a8084bc54c2fbd6b0bd235498.ptp	2022/7/23 18:38	PTP ファイル	128 KB
10-74c06ce4b50e1038a0614dd973bb0b26a62da6862cd484423e355fd74f4df69b.ptp	2022/7/23 18:38	PTP ファイル	128 KB
11-99b12d2285d96025422f1c57561aa1941a5b3c0f83f871d3df23268bdfdfc8fc.ptp	2022/7/23 18:38	PTP ファイル	128 KB
12-11c28cf53791d1866c70efd6a4d387c50a9a68ba20ae8e711aab5cada39a8bda.ptp	2022/7/23 18:38	PTP ファイル	128 KB
13-a1791c039ca251d60f4af743cd9c1998a9c24deeb687de7bdbb39fe2d33ab3e5.ptp	2022/7/23 18:38	PTP ファイル	128 KB
14-8d3cf07603c286e8395d604f654e83e06b8b416e982544c6011362c8df9c87ee.ptp	2022/7/23 18:38	PTP ファイル	128 KB
15-e90609ad53449aaa7d0a330f1e87c2bd5e4e318e647b35339ece921b04b15165.ptp	2022/7/23 18:38	PTP ファイル	128 KB
16-4ed5f9f31abe3e22b39ef71c9bc44c7f54204bc9310df8cb62977687826e748e.ptp	2022/7/23 18:38	PTP ファイル	128 KB
17-d0fa962db5a49fb7f29e6e1ea9311642de29af9072ba49b9b342101909e92509.ptp	2022/7/23 18:38	PTP ファイル	128 KB
18-c755955f79af415422a7d08267f80ec712f2333409fa978c2d9c7f98702349ca.ptp	2022/7/23 18:38	PTP ファイル	128 KB
19-44fef7fc593dfcd0eb0373c81525d045d5de558a00676aaff7682a88919ebf82.ptp	2022/7/23 18:38	PTP ファイル	128 KB
20-1649576ad107e784730214dcd86fc133057d0af1782d2cfb6c80cf4d392c04a3.ptp	2022/7/23 18:38	PTP ファイル	112 KB

splits

3.2 檔案的上傳

分割完成後，每個檔案將產一個 File ID（SHA-256 雜湊值）和 split 目錄（包含所有 split 的名稱的目錄）。這些資訊及 Node Address 將透過 PT-0105 協定一併被上傳到 Node 指定的 Centers，而 Center 接收到這些資訊後便會將其儲存在系統中，供其他節點查詢。（注意，splits 將不會被上傳到 Center）

3.3 種子檔（Seed）的生成

檔案相關的資訊上傳至 Center 之後，Center 將返回 PT-0205 代碼告知節點上

傳已經完成。當節點收到所有來自 Center 的訊息後，便會將上傳成功的節點目錄以及 split 目錄置於種子檔並將其匯出。種子檔的格式如下：

```
{
    "FILE_ID": "0e8dfa0194052893e3881dae768a92e19e212fbcbe5d4521da7e84b7998fabbef
    "FILE_NAME": "[redacted]
    "CENTERS": [
        "2643xvxgfw5yzo2m7i5j6hxp44nyq3kzp3zzi26y3q7x4ei2kr5dbaad.onion",
        "-pimojbov6ds2ayuivtaxb5hgqvxvlfdxodmznvj2giakyvk57le2psyd.onion"
    ],
    "SPLITS": [
        "0-d5c00c10d05c416c546c8d676a0e2716b73ca99d4a47e42e37a232ebb2cac08c",
        "1-467041f0b12e500500cffa6ce52e235ec44c1c631ecade7357ca8feabe73a711",
        "2-ede1ca484d99bb86d6a380fb28a6faed10fceea925466b398fa513ed0ab6ecd5",
        "3-b7448ddd3b5870c186f761bb0b52e2ad687a61cbdc738627faa95f337b248ca9",
        "4-b9aea19e3cfa8df7010ba2aac97780aeedbd247b96ba8f1367f6a5c25eac3391",
        "5-786b590b2a6543584b8b09cad4bcdd7dd648f763ae22b8ec5a1095005f5f395b",
```

種子檔之標準格式（部分）

其中，目錄「CENTERS」為該檔案的資訊目前位於哪些 Centers，而下載節點可以向這個目錄上的任意 Center 要求檔案的 splits 之分布位置。而目錄「SPLITS」則告訴下載節點有哪些 splits 必須下載，並且節點可以依據這個目錄得知本機目前缺少哪些 splits。

3.4 檔案的下載

真正開始下載一個檔案之前，必須先取的該檔案的種子檔。若缺少種子檔，則無法得知 File ID 及其所在位置。取得種子檔後，Node 便會以迭代的方式向種子檔內「CENTERS」目錄上的 Centers 要求 splits 位置資訊，若其中一個 Center 回應了 Node 的請求，則不再向下一個 Center 發送訊息。以下是 Node 成功向 Center 要求的資訊範例：

```
{
    "7k3secb7selqspynfrzoujdp2e6ofyl7vavw6efp4rubwmkjxz5ndoyd.onion": [
        "0-d5c00c10d05c416c546c8d676a0e2716b73ca99d4a47e42e37a232ebb2cac08c",
        "1-467041f0b12e500500cffa6ce52e235ec44c1c631ecade7357ca8feabe73a711",
        "2-ede1ca484d99bb86d6a380fb28a6faed10fceea925466b398fa513ed0ab6ecd5",
        "3-b7448ddd3b5870c186f761bb0b52e2ad687a61cbdc738627faa95f337b248ca9",
        "4-b9aea19e3cfa8df7010ba2aac97780aeedbd247b96ba8f1367f6a5c25eac3391",
        "5-786b590b2a6543584b8b09cad4bcdd7dd648f763ae22b8ec5a1095005f5f395b",
        "6-37b26b4f57ed6a33d08632b428cab2a7e675d98ee4233795de8fcb1f92eed3df",
        "7-9581c9a5aa1be39753ee0092ef33773889135cbc7da7768fb1cf79ed3f4c0fb6",
        "8-731e84025fce6569057f45c5c95838c2d32909bb21c222053ac57b61f305b09c",
        "9-fa127cceacab85480e18e30d4abcca3bcbb0ae8a8084bc54c2fbd6b0bd235498",
        "10-74c06ce4b50e1038a0614dd973bb0b26a62da6862cd484423e355fd74f4df69b",
        "11-99b12d2285d96025422f1c57561aa1941a5b3c0f83f871d3df23268bdfdfc8fc",
        "12-11c28cf53791d1866c70efd6a4d387c50a9a68ba20ae8e711aab5cada39a8bda",
        "13-a1791c039ca251d60f4af743cd9c1998a9c24deeb687de7bdbb39fe2d33ab3e5",
        "14-8d3cf07603c286e8395d604f654e83e06b8b416e982544c6011362c8df9c87ee",
        "15-e90609ad53449aaa7d0a330f1e87c2bd5e4e318e647b35339ece921b04b15165",
```

Node 向 Center 要求的 splits 位置資訊（部分）

收到這些資訊後，Node 將進行下一步 — 下載安排（Downloading Arrangement）。

下載安排，即決定該向哪一個 Node 下載哪一個 split 的安排。Node 使用以下演算法來完成下載安排：

```
download_arrangement ← empty array
node_weights_counter ← empty array
split_counter ← empty array

// Part A
foreach node_address, split in node_info_from_center
    split_counter[split] += 1
    node_weights_counter[node_address] += 1

// Part B
sort(split_counter)
reverse(node_address)

// Part C
while split_counter is not empty
    temp_node_weights_counter ← node_weights_counter
    foreach split in split_counter
        foreach node in temp_node_weights_counter
            download_arrangement[node] ← split
            delete split from split_counter
            delete node from temp_node_weights_counter
            break
```

用於下載安排的演算法

Part A：計算權重

Node 將計算每個檔案來源（擁有檔案之 Node）及每個 split 的權重。檔案來源的權重計算以該來源傭有多少 splits 決定，擁有 splits 數量愈多者權重愈大；split 的權重則以該 split 在整個網路中有多少數量決定，數量愈少權重愈大。

Part B：權重排序

完成計算後，檔案來源及 splits 將以上述規則進行權重排序。

Part C：下載安排

下載將向有最多 splits 的 Node 下載數量最少的 split。這確保數量最少的 split 不會消失於 PTP 網路，並且擁有最完整檔案的 Node 將會被請求最多次。

4. PTP 傳輸協定

4.1 標準格式

PTP 傳輸協定具有以下標準格式：

```
CONNECTOR :
CONNECTION_CODE :
DATA :
```

PTP 傳輸協定標準格式

CONNECTOR：連線來源。PTP 所有的連線皆透過 Tor 進行，因 Tor 的特性，被連線端無法得知連線來源，因此連線端必須在連線資訊內聲明連線來源。這有一個風險，那就是 Tor 出口節點可以透過監聽得知哪兩個節點正在連線，不過因為 Node 之間是透過完全匿名的 address 相互識別，因此除了這些資訊，攻擊者無法進一步鎖定個人。

CONNECTION_CODE：通訊碼。透過通訊碼，節點可以判斷該連線的目的，並做出進一步的反應。關於通訊碼的細節將在下一節做更完整的說明。

DATA：通訊內容。通訊內容並無一定格式，各通訊碼所附帶之 DATA 值皆不同。

4.2 通訊碼

所有通訊碼皆以「PT-」開頭。而其後四位數之前兩位代表通訊端的種類：01 表示「Node to Center」；02 表示「Center to Node」；03 表示「Node to Node」；04 表示「Center to Center」。

4.2.1 PT-01XX

PT-0105：向 Center 上傳 seed 資訊

當 Node 上傳新的檔案時（參見 3.2 檔案的上傳），使用 PT-0105 通訊碼向 Center 傳送 seed 資訊。

PT-0106：向 Center 取得當前 seed 資訊

Node 下載檔案前，向 Center 取得該檔案的 splits 位置資訊。

PT-0109：向 Center 更新 seed 資訊

當下載節點向來源節點成功下載 split 後，來源節點將發送 PT-0109 代碼向 Center 告知，下載節節點已成功取得該 split。

4.2.2 PT-02XX

PT-0205：回傳 seed 資訊上傳狀態

當 Center 接收到 PT-0105 代碼後，便將相關資料寫入資料庫建立。而檔案上傳節點則透過該代碼確認檔案是否以成功寫入資料庫。若成功則將該節點放入 seed 中，反之，則略過該 Center。

PT-0206：回傳當前 seed 資訊

當 Node 取得種子檔後，向種子檔內的 Center 目錄要求 seed 資訊（參見 3.4 檔案的下載）時，Center 回傳 seed 資訊時所使用的代碼。

4.2.3 PT-03XX

PT-0301：要求 split

下載節點向來源節點要求 split。

PT-0302：發送下載成功訊息

下載節點向來源節點發送下載成功的訊息，讓來源節點向 Centers 更新 seed 資訊。

PT-0307*：發送 split（不會回傳代碼）

當下載節點成功向來源節點請求 split 後，來源節點返回要求之 split。所有 splits 皆會以 AES-256CBC 加密。

PT-0308：拒絕下載要求（已棄用）

PT-0309.1：要求加密密碼及 IV

在下載 splits 前，下載節點會將 public key 傳送給來源節點，並向來源節點要求該次加密的 password 及 IV。

PT-0309.2：返回加密密碼及 IV

當來源節點收到 PT-0309.1 要求時，會產生一組 password 和 IV，這將用於接下來的檔案傳輸。

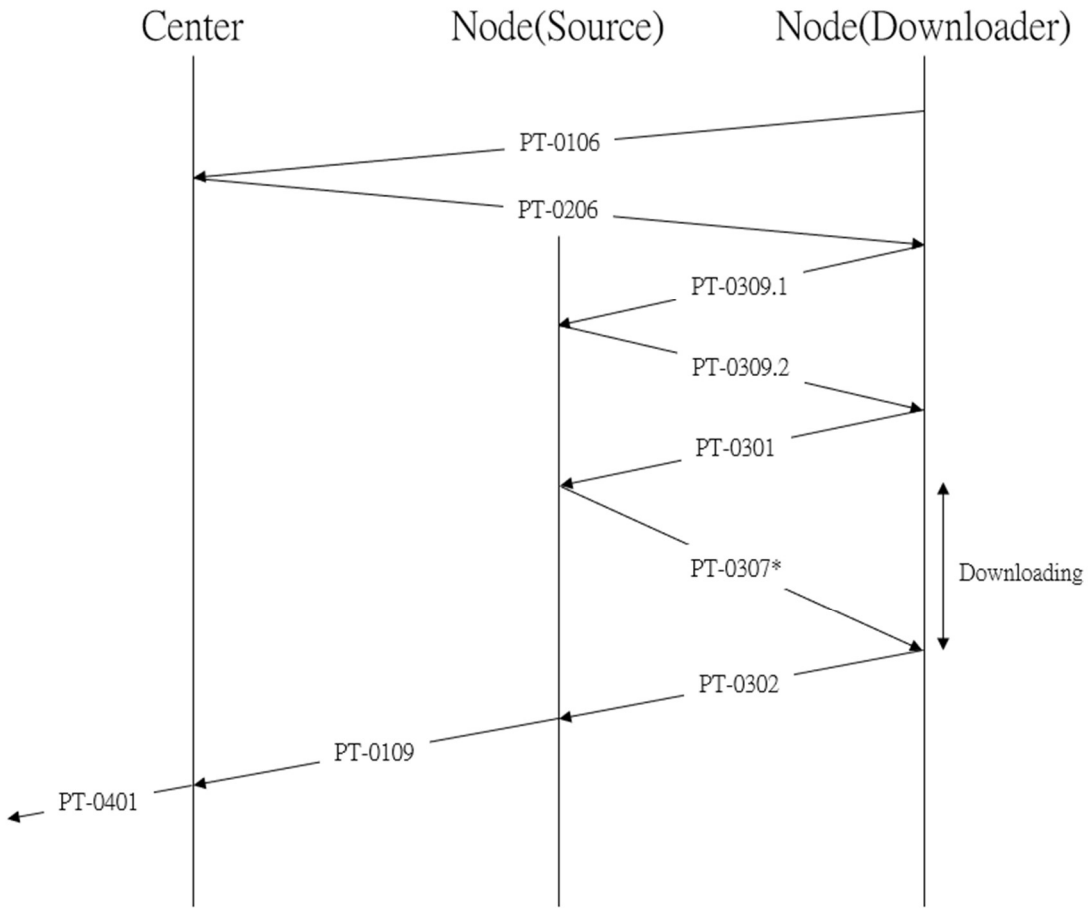
5.2.1 PT-04XX

PT-0401：共享 seed 資訊

當 Center 的 seed 資訊被更新時，系統會同步把這項更新廣播到所有其他信任的 Centers（本機節點列表中之 Centers）。

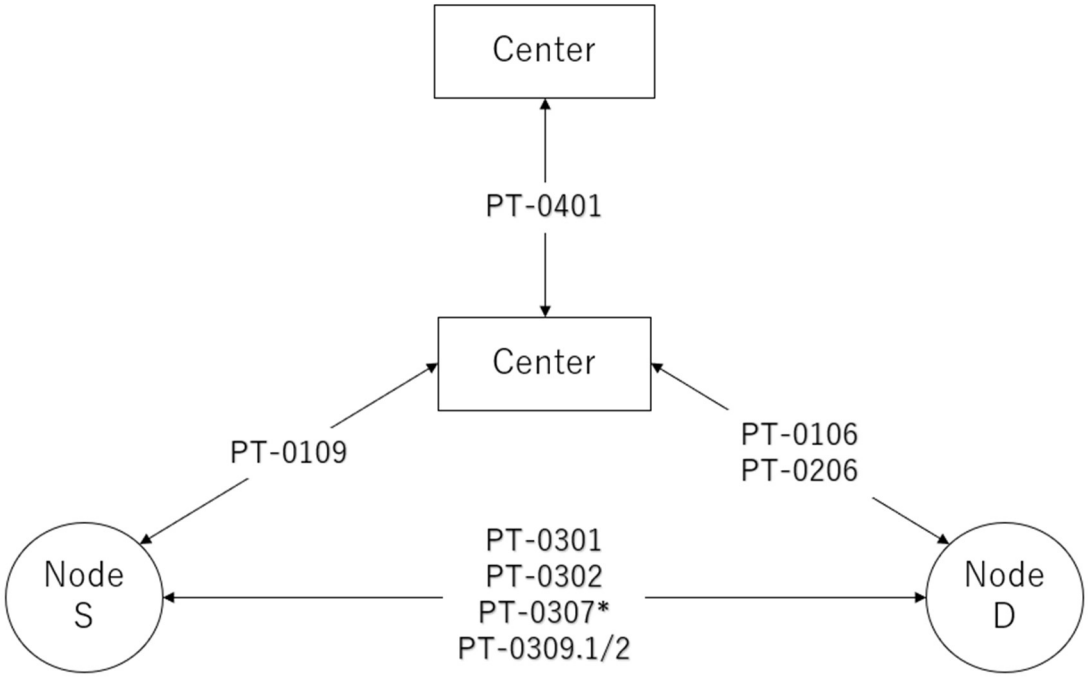
4.3 通訊流程

整個 PTP 網路的連結皆是通過 PTP 通訊協議完成。下面的示意圖為下載節點從一開始向 Center 請求 seed 資訊到下載完成後來源節點向 Center 更新 seed 資訊的時序圖：



通訊時序圖（忽略節點處裡時間）

首先，下載節點（以下稱 Node D）向 Center 請求 seed 資訊（PT-0106），接收資訊（PT-0206）後，Node D 便開始向 seed 之節點列表中的節點傳送 public key（PT-0309.1），請求檔案加密密碼及 IV，若來源節點（以下稱 Node S）接受連線，則返回經 public key 加密的檔案加密密碼及 IV（PT-0309.2）。此一步驟確保縱使連線被監聽，監聽者也無法得知檔案內容。收到該次傳輸所用之加密密碼及 IV 後，Node D 接著便繼續向 Node S 請求 splits（PT-0301），而 Node S 收到請求後，便將檔案加密，傳送給 Node D（PT-0307）。當 Node D 收到後，便以先前處存之密碼及 IV 將其解密並加以驗證，並回傳確認訊息（PT-0302）告知已成功下載。



Node 及 Center 關係圖

5. 使用手冊

5.1 檔案說明

Node~ /index.php：Node 運行的主要檔案。

Node~ /run.bat：Node 啟動檔。

Node~ /conf/：設定檔目錄。

Node~ /conf/centers.json：本機 Center 目錄。用戶可將信任的 Center 加入此目錄，所有 seed 相關資料將上傳至此目錄中的 Center。

Node~ /conf/config.json：本機設定檔。包含 Node address、Public key，以及 Private key。

Node~ /conf/nodes.json：遠端 Node 連線資料，包含遠端 Node address、Public key、檔案加密密碼，以及 IV

Node~ /ptp.ini：本機設定檔。

Node~ /download/：下載檔案預設存放位置。

Node~ /src/：The source directory.

Node~ /src/async_connect.php：非同步連線檔。用於實現非同步資料傳輸功能。

Node~ /src/functions.php：主功能檔。

Node~ /src/onion_generator.php：Node address 生成檔。僅作初始化 Node 用。由 The Tor Guy tordevstuff@protonmail.com 授權提供。

Node~ /seed/：分割檔（split）儲存用資料夾。

Node~ /temp/：Seed 資料儲存資料夾。用於儲存從 Center 取得的當前 Seed 資訊。

Center~ /centers.json：本機 Center 目錄。用戶可將信任的 Center 加入此目錄，當 database 內的 seed 資訊被更新時，所有更新皆會被廣播至目錄內所有節點。

Center~ /database/n：Seed 資訊儲存資料夾。

5.2 常見問題

1. 如何建立私人 PTP 網路？

私人 PTP 網路可以透過架設私人 Center 實現。Address 幾乎不可能被猜測，因此若不向不被信任的第三者得知私人 Center 的 address，則該 PTP 網路可以稱為私人的。

2. 如何變更 public/private key pair？

Key pair 的更新很簡單，若 private key 不慎被其他人得知，則可以透過刪除 /conf/config.json 內的 PUBLIC_KEY 和 PRIVATE_KEY 即可。（注意，function initialize() 必須被啟用）。