多段プロキシによる
Tor の Exit ノードの隠蔽について

ＮＴＴ コミュニケーションズ株式会社
経営企画部
マネージドセキュリティサービス推進室
セキュリティオペレーション担当

2013 年 03 月 15 日

Ver. 1.0

1. 調査概要

1.1. 調査概要

Tor 経由でアクセスしているかどうかは、Tor の Exit ノードの IP アドレスかどうかをチェックすることである程度の判別は可能だ。Tor 経由で Open Proxy を使うことで、Tor の Exit ノードの IP アドレスをサーバ側のログに残さない方法が可能かどうか調査した。
結論として、この方法によって、サーバ側に Tor を使っているかどうかを隠蔽することができた。

2. 注意事項

Tor は元々、言論の自由のない弾圧された国家や社会などで暮らす人々へ匿名でも発言できるように、また(国家権力の犯罪行為など)巨悪を告発する際の告発者のプライバシーを保護するために作られたもので、犯罪者が身元を隠すためのツールではない。
当然だが、本文書の内容を悪用することは厳禁とする。
本文書の内容が、言論の自由を奪われ弾圧されている人々の役に立てれば、個人的に幸いである。

3. 検証結果

実際に検証した結果、特に障害となるような事はなく、実現可能であることを確認した。
また、作成したツールは、元々この検証を目的として作成されたわけではなく、あくまでネットワーク試験を行うために作成したものを改造している。
最後に、もう一度記すが、「2 注意事項」でも記しているが、本文書の内容を悪用することは厳禁である。

3.1. 検証環境

図 3.1-1のように、通信をSOCKSへ転送可能なプログラムを作成し、そこからTorを経由し、Open Proxyを経由し、Webサーバ(クライアントのIPアドレスを表示するCGI)へアクセスするという経路をとる。

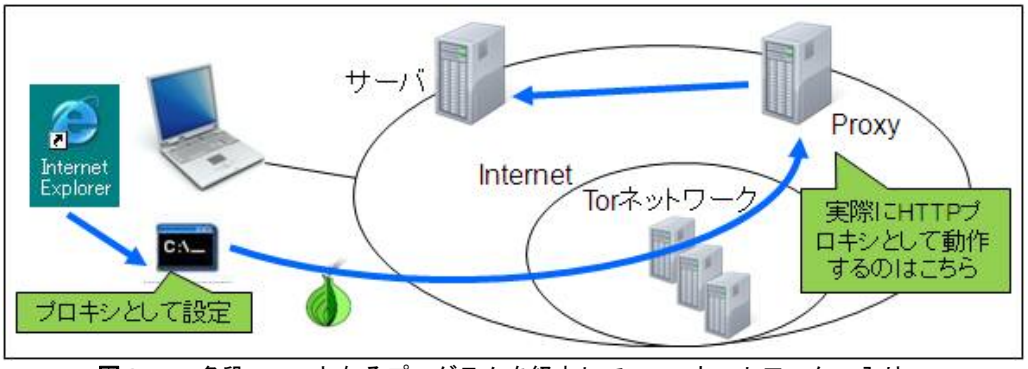

図 3.1-1 : 多段 Proxy となるプログラムを経由して、Tor ネットワークへ入り、その後 Open Proxy からサーバへアクセスする

接続先サーバ	www.example.com
SOCKS サーバ	192.0.2.5
Open Proxy Server1	XXX.XX.XX.XX
Open Proxy Server2	ZZZ.ZZZ.ZZ.ZZ

図 3.1-2 : 検証で用いたサーバのアドレス

SOCKS サーバは、Tor への入り口となるサーバである

3.2. 検証 1

「SOCKS サーバ」→「Tor」→「Open Proxy Server1」→「接続先サーバ」という順で接続してみた。特に問題もなく、接続に成功した。また、接続先サーバには、「Open Proxy Server1」からの接続のように見えることも確認した。

```
C:¥tcpRelay>TcpRelay.bat -Verbose -LocalPort 0 -RemotePort 80 -RemoteHost www.exampl
e.com -proxy socks5://192.0.2.5:9050 -proxy connect://XXX.XX.XX.XX:8080
TcpRelay for JAVA ver 2.0
        create by active@window.goukaku.com

                                        ～省略～

-------- Configuration Infomation --------
  Local is stdin/out
  Redirect is 192.0.2.5:9050
  Binary mode
  Verbose Mode on
Proxy Setting is
     socks5://anonymous@XXX.XX.XX.XX:8080
     connect://anonymous@www.example.com:80
  Thread Interval time is 500(ms)
  No Limit Connection
stdin/stdout => 192.0.2.4:1063 -> 192.0.2.5:9050
Connected: XXX.XX.XX.XX:8080 (socks5)
Connected: www.example.com:80 (connect)
GET /r_host.asp HTTP/1.0                        ← キーボードから入力

HTTP/1.1 200 OK
Connection: close
Date: Thu, 07 Mar 2013 15:26:58 GMT
Server: Microsoft-IIS/5.2 SP3 rc14 Beta
X-Content-Type-Options: nosniff
X-Powered-By: ASP.NET
Content-Type: text/html; charset=shift_jis
Set-Cookie: ASPSESSIONIDAARBACCT=IDFDJNAAEGLELLAEGOIHJELI; path=/; httponly
Cache-control: private
X-Powered-By: sISAPILocation 1.0.2.2


<HTML>
 <HEAD>
  <TITLE>あなたの情報</TITLE>
  <LINK REL="SHORTCUT ICON" HREF="/favicon.ico">
 </HEAD>
 <BODY>
  <UL>
   <LI>Your HostName : XXX.XX.XX.XX<LI>IP Address : XXX.XX.XX.XX<LI>YourBrowser: <TA
BLEBORDER="1"><TR><TD>REMOTE_HOST</TD><TD>XXX.XX.XX.XX</TD></TR><TR><TD>REMOTE_ADDR<

                                        ～省略～
```

図 3.2-1：多段 Proxy となるプログラムを経由して、Tor ネットワークへ入り、その後 Open Proxy からサーバへアクセスした結果

```
C:¥tcpRelay>TcpRelay.bat -verbose -localport 92 -remoteport 8080 -remoteHost XXX.XX.
XX.XX -proxy socks5://192.0.2.5:9050
TcpRelay for JAVA ver 2.0
        create by active@window.goukaku.com

                                        ～省略～

-------- Configuration Infomation --------
  Local Port = 92
  Redirect is 192.0.2.5:9050
  Binary mode
  Verbose Mode on
Proxy Setting is
     socks5://anonymous@XXX.XX.XX.XX:8080
  Thread Interval time is 500(ms)
  No Limit Connection
127.0.0.1:1071 -> 127.0.0.1:92 => 192.0.2.4:1072 -> 192.0.2.5:9050
Connected: XXX.XX.XX.XX:8080 (socks5)
バッチ ジョブを終了しますか (Y/N)? y
```

図 3.2-2：図 3.2-1と同じ設定で、92/tcpで待機する

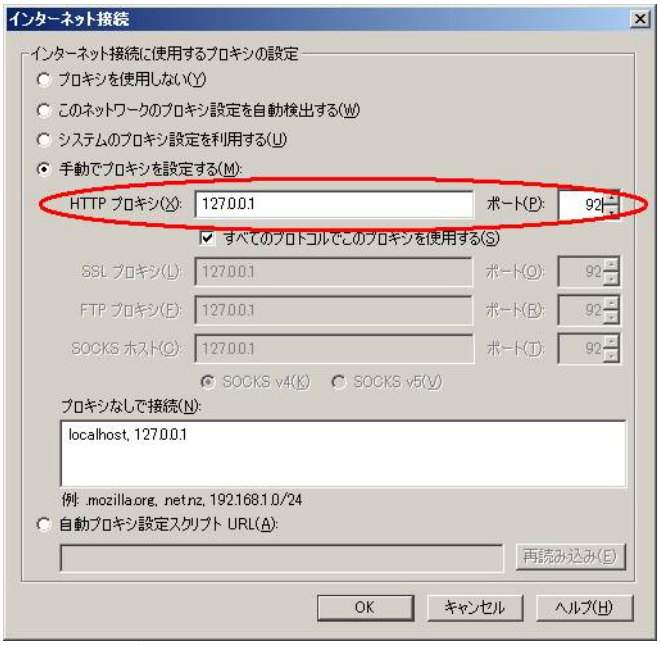

図 3.2-3：Webブラウザの Proxy設定を図 3.2-2に接続するようにする

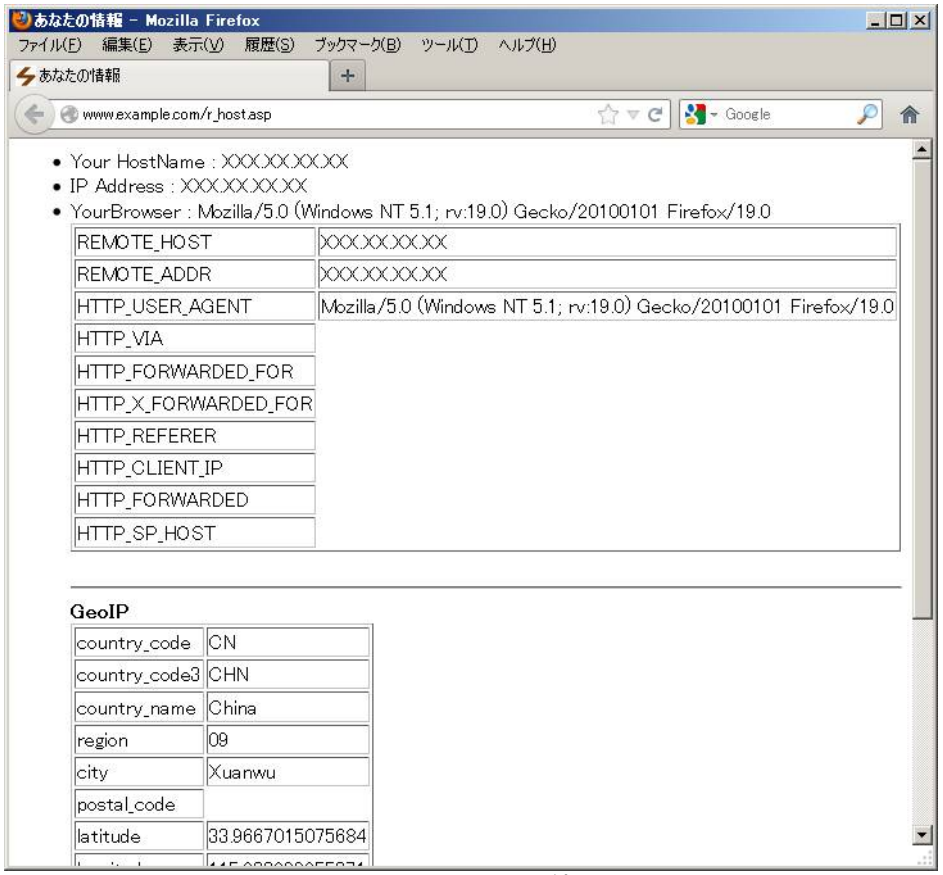

図 3.2-4 : 図 3.2-3の結果

「Tor」→「Open Proxy Server1」経由で接続することができた

3.3. 検証 2 (Open Proxy 自体を多段に設定する)

「SOCKS サーバ」→「Tor」→「Open Proxy Server1」→「Open Proxy Server2」→「接続先サーバ」という順で接続してみた。
特に問題もなく、接続に成功した。また、接続先サーバには、「Open Proxy Server2」からの接続のように見えることも確認した。

```
C:¥tcpRelay>TcpRelay.bat -verbose -localport 0 -remoteport 80 -remotehost
ZZZ.ZZZ.ZZ.ZZ -proxy socks5://192.0.2.5:9050 -proxy connect://XXX.XX.XX.XX:8080
TcpRelay for JAVA ver 2.0
        create by active@window.goukaku.com

                                        ～省略～

-------- Configuration Infomation --------
  Local is stdin/out
  Redirect is 192.0.2.5:9050
  Binary mode
  Verbose Mode on
Proxy Setting is
     socks5://anonymous@XXX.XX.XX.XX:8080
     connect://anonymous@ZZZ.ZZZ.ZZ.ZZ:80
  Thread Interval time is 500(ms)
  No Limit Connection
stdin/stdout => 192.0.2.4:1144 -> 192.0.2.5:9050
Connected: XXX.XX.XX.XX:8080 (socks5)
Connected: ZZZ.ZZZ.ZZ.ZZ:80 (connect)
GET http://www.example.com/r_host.asp HTTP/1.0              ← キーボードから入力

HTTP/1.0 200 OK
Date: Thu, 07 Mar 2013 15:56:47 GMT
Server: Microsoft-IIS/5.2 SP3 rc14 Beta
X-Content-Type-Options: nosniff
X-Powered-By: ASP.NET
Content-Type: text/html; charset=shift_jis
Set-Cookie: ASPSESSIONIDAARBACCT=BEFDJNAALABELHLEENNKJFOH; path=/; httponly
Cache-Control: private
X-Powered-By: sISAPILocation 1.0.2.2
Connection: close


<HTML>
 <HEAD>
  <TITLE>あなたの情報</TITLE>
  <LINK REL="SHORTCUT ICON" HREF="/favicon.ico">
 </HEAD>
 <BODY>
  <UL>
   <LI>Your HostName : ZZZ.ZZZ.ZZ.ZZ<LI>IP Address : ZZZ.ZZZ.ZZ.ZZ<LI>YourBrowse
r : <LI>ProxyServer : <LI>Your Real IP Address : XXX.XX.XX.XX<TABLE
BORDER="1"><TR><TD>REMOTE_HOST</TD><TD>ZZZ.ZZZ.ZZ.ZZ</TD></TR><TR><TD>REMOTE_ADDR

                                        ～省略～
```

図 3.3-1：多段 Proxy となるプログラムを経由して、Tor ネットワークへ入り、

その後 Open Proxy からサーバへアクセスする

```
C:¥tcpRelay>TcpRelay.bat -verbose -localport 92 -remoteport 80 -remotehost
ZZZ.ZZZ.ZZ.ZZ -proxy socks5://192.0.2.5:9050 -proxy connect://XXX.XX.XX.XX:8080
TcpRelay for JAVA ver 2.0
        create by active@window.goukaku.com

                                    ～省略～

-------- Configuration Infomation --------
  Local Port = 92
  Redirect is 192.0.2.5:9050
  Binary mode
  Verbose Mode on
Proxy Setting is
     socks5://anonymous@XXX.XX.XX.XX:8080
     connect://anonymous@ZZZ.ZZZ.ZZ.ZZ:80
  Thread Interval time is 500(ms)
  No Limit Connection
127.0.0.1:1141 -> 127.0.0.1:92 => 192.0.2.4:1142 -> 192.0.2.5:9050
Connected: XXX.XX.XX.XX:8080 (socks5)
Connected: ZZZ.ZZZ.ZZ.ZZ:80 (connect)
バッチ ジョブを終了しますか (Y/N)? y
```

図 3.3-2 : 図 3.3-1と同じ設定で、92/tcpで待機する

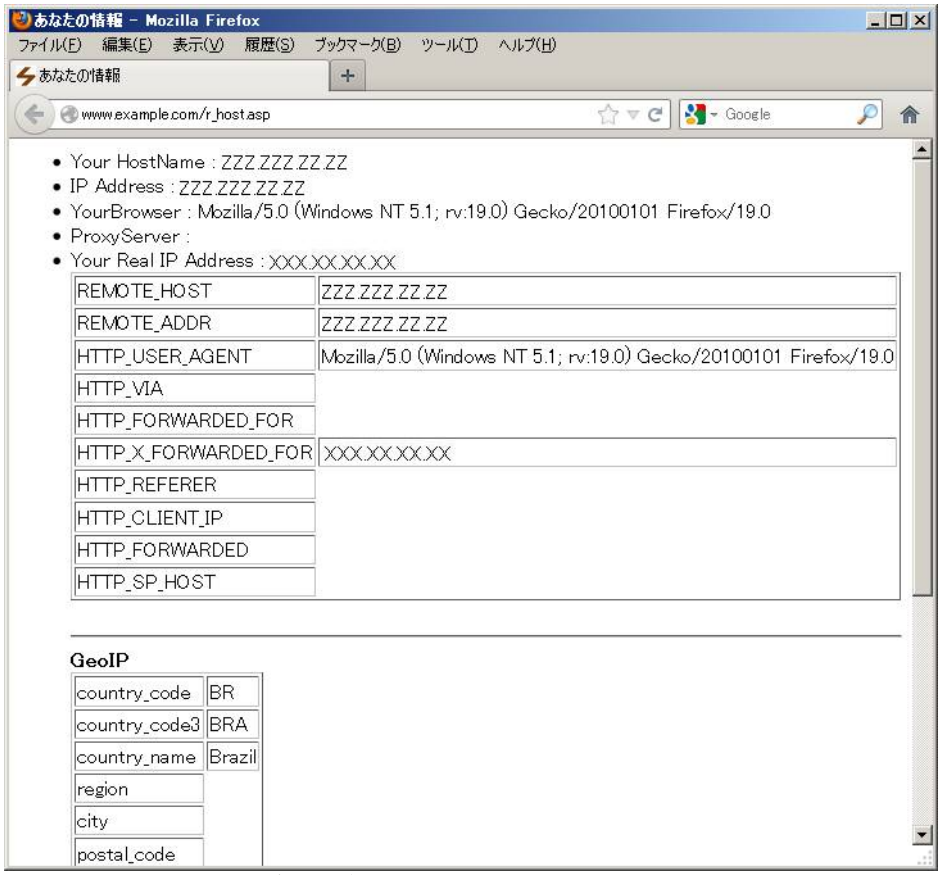

図 3.3-3 : WebブラウザのProxy設定を図 3.3-2に接続するようにして接続先サーバへアクセスした結果。「Tor」→「Open Proxy Server1」→「Open Proxy Server2」経由で接続することができた

4. 検証作業者

NTT コミュニケーションズ株式会社
経営企画部マネージドセキュリティサービス推進室
セキュリティオペレーション担当
　佐名木 智貴

5. 履歴

- 2013 年 03 月 15 日 ：ver1.0 最初の公開

6. 最新版の公開URL

http://www.ntt.com/icto/security/data/soc.html

7. 参考

- 「Web ブラウザの SOCKS 実装状況について」
  http://www.ntt.com/icto/security/data/soc.html
- 「Tor の安全な使用方法について(DNS Leakage の観点から)」
  http://www.ntt.com/icto/security/data/soc.html
- SOCKS
  http://ja.wikipedia.org/wiki/SOCKS
- SOCKS: A protocol for TCP proxy across firewalls
  http://ftp.icm.edu.pl/packages/socks/socks4/SOCKS4.protocol
- RFC1928
  http://tools.ietf.org/html/rfc1928
- RFC1929
  http://tools.ietf.org/html/rfc1929
- RFC1961
  http://tools.ietf.org/html/rfc1961
- Tor Project
  https://www.torproject.org/
- TCP Relay for Java
  http://rocketeer.dip.jp/sanaki/free/jfree11.htm

8. 本レポートに関する問合せ先

NTT コミュニケーションズ株式会社
経営企画部
マネージドセキュリティサービス推進室
セキュリティオペレーション担当

e-mail: scan@ntt.com

以 上