UDP-timeout algorithms used in various NAPT routers

HAYASHI yu-ichi
Core and Information Technology Center

1 はじめに

NAPT を利用したネットワーク環境の中で外部とのコネクションを TCP/IP を利用して確立する場合, NAPT は NAPT の外側のポートと内側にいるホストの IP をマッピングしプライベートネットワークとグローバルネットワークのコネクタビリティを保つ機能を担っている. NAPT は内部のホストが外側のネットワークに通信を行う際, 先のマッピングを行うとともに, それらのマッピングにタイマーを掛け, ある一定時間以上通信が行われない場合にマッピングを解除する仕組みを有している. これをここでは NAPT タイマーと呼び, 各実装による NAPT タイマーの時間を計測し, それらの動きを推察する. なおこれらのアドレスとポートのマッピングとタイマーの組み合わせにより, プライベートな空間が一時的にグローバル空間とつながりこの間 NAPT におけるセキュリティの脆弱性が見られることも予想する

2 実験環境

NAPT を介してプライベートネットアドレスを保有するホスト(Ph)とグローバルアドレスを保有するホスト(Gh)との通信を, UDP を用い行い NATP のアドレスとポートのマッピングを観察し, 通信時間におけるそれらの解放を行うタイマーによるマッピングの状態遷移を観察する(図1)

3 アドレスとポートのマッピングと解除

この実験構成で利用した NAPT は OpenBlocks[1], CR20, CR110[2] である. 各 NAPT に対して上記の実験構成を行い NAPT タイマーにおける状態遷移を調べた

行った実験手順は次の通りである.

手順1: Ph から NAPT を介して Gh に対して接続要求を行い Ph から要求を受けた Gh は一定の間隔(T1)を置いてから Ph に接続確認を送信する

手順2: 接続確認を受けた Ph は Gh に反応を返さず, さらに一定期間(T2)をおいて Gh が確認要求を Ph に送信する.

手順3: これらを続け, Ph が Gh からの確認要求をタイマーによる制御により受け取れなくなるまでこれらを監視する.

この実験方法において TCP は用いず UDP を利用して実験した.

3.1 OpenBlocks

手順1・手順2を行った結果30秒間 Gh からのレスポンスがないとタイマーはマッピングを解除する(図2)

手順2を30秒以内で行いマッピングが解除される前に次の確認要求を Gh から送信すると180秒間 Gh からの確認要求が行われないとタイマーはマッピングを解除する(図3)

手順1, 手順2(Gh から30秒以内に確認要求を送信), 手順3(Gh から180秒以内に確認要求を送信)したあとにさらに手順3を続けて行った場合もまた180秒以内に確認要求が行われないとタイマーはマッピングを解除する

この NAPT でアドレスとマッピングされるポート番号は1026番から5000番までの3975個で, 各 Ph ホストからの外部からの接続要求の合計が3975個を超えた場合 NAPT を利用したそれ以降の通信はマッピングの解除を待ってから行われる. また, 接続要求がだされてから30秒以内はタイマーはアドレスのマッピングを解除しないため, 30秒以内に3975個以上の接続要求を発生させると他の通信が行えなくなるが, 同アドレスから30秒以内に3975個以上の接続要求を出した場合そのアドレスは NAPT よりロックされ以後しばらくの間通信を行うことができなくなる

3.2 CR20・CR110

手順1・手順2を行った結果60秒間 Gh からのレスポンスがないとタイマーはマッピングを解除する(図4)

手順2を60秒以内で行いマッピングが解除される前に次の確認要求を Gh から送信すると30秒間 Gh からの確認要求が行われないとタイマーはマッピングを解除する(図5)

手順1, 手順2(Ghから60秒以内に確認要求を送信), 手順3(Ghから30秒以内に確認要求を送信)したあとにさらに手順3を続けて行った場合もまた30秒以内に確認要求が行われないとタイマーはマッピングを解除する

この NAPT でアドレスとマッピングされるポート番号は12288番から32767番までの20480個で, こちらは60秒以内に 1 つのホストからこれだけソケットを生成することは非常に困難であり, OpenBlocks にみられた手法で NAPT の内部にあるプライベート空間からの通信を停止させることな非常に困難であるが, 複数のホストから接続要求を 60 秒以内に20480個発生させればそれ以上の通信は行えず, また先にみられた OpenBlocks のようなアドレスのロックはみられない.

4 タイマーと状態遷移

NAPT は Ph が最初に通信の接続要求を Gh に送信してから接続確認を受信するまでに送信(sent)と 1 回目の確認要求の受け取り(First Recv)という2つの状態遷移があり, それ以後は確認要求を受け付ける状態(Recv)という計3つの状態遷移を持っていることがわかる(図6).

5 タイマーの最適化と動的制御

これらのタイマーは各ベンダーの実装により一定ではなく, 外部からこの状態遷移を知ることができればアドレスのマッピングが解除される前に内部に接続することが可能である。NAPT は内部の通信状況に合わせてタイマーを制御し変更する実装を設け, それを利用し外部からのタイマーの予測と攻撃を避けるように設計されるべきである。

6 おわりに

各家庭や, 企業が NAPT を導入しそれらにより外部からの接続を遮断できるという考えはあくまで内部アドレスは外部に告知されず, ルーティングされないという観点からであり, 接続が行われないというものではない。NAPT を利用する際には必ずフィルタ等を同時に利用し, そういった攻撃に備えるべきである。

参考文献

[1] OpenBlocks:
http://online.plathome.co.jp/products/openblocks/openblockss/
[2]CR20, CR110:
http://www.centurysys.co.jp/product/index.html

図1:NAPTタイマー実験環境

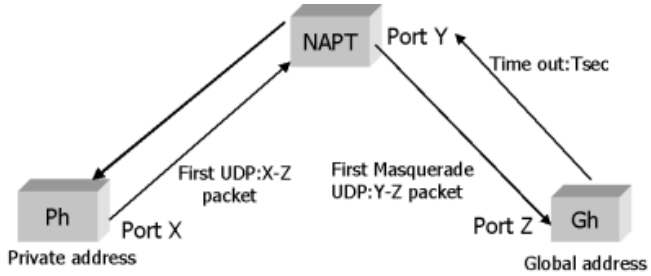

図2:手順1, 手順2における状態遷移 (OpenBlocks)

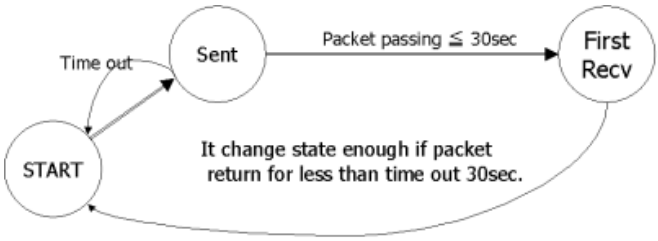

図3:手順1, 手順2, 手順3, における状態遷移(OpenBlocks)

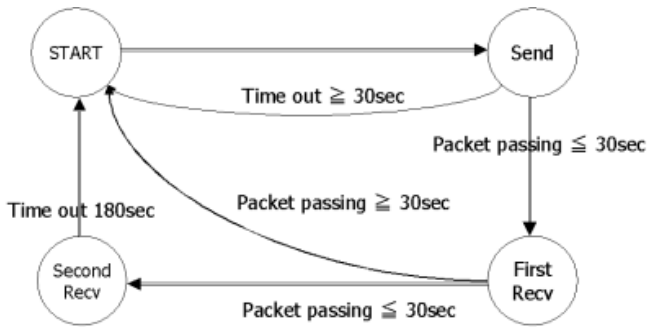

図4:手順1, 手順2における状態遷移 (CR Series)

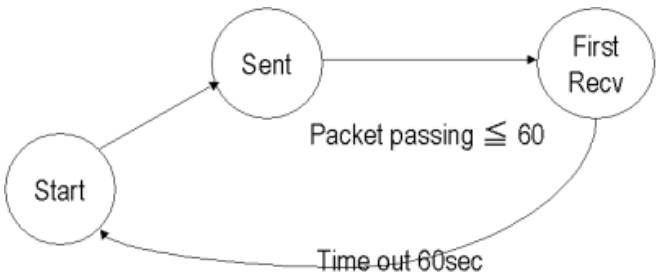

図５：手順 1，手順 2，手順３，における状態遷移 (CR Series)

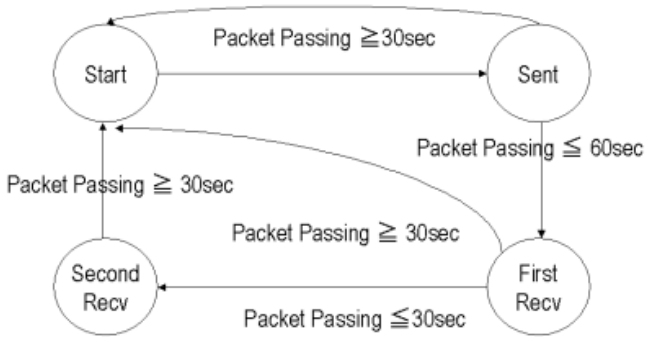

図6:状態遷移(まとめ)