

Liste des principales commandes CMD

Gestion des réseaux et Internet

- **CONTROL NETCONNECTIONS** ou bien **NCPA.CPL** : ouvre les connexions réseau
- **FIREWALL.CPL** : ouvre le pare-feu Windows
- **INETCPL.CPL** : ouvre les propriétés internet
- **IPCONFIG** : affiche les configurations des adresses IP sur l'ordinateur (Pour plus d'informations, tapez **IPCONFIG /?** dans l'invite de commande **CMD**)
- **NETSETUP.CPL** : ouvre l'assistant configuration réseau (Pour XP uniquement)
- **WF.MSC** : ouvre les fonctions avancées du pare-feu Windows (Pour Vista uniquement).
Tutoriel [ICI](#)
- À VOIR ÉGALEMENT : [Commandes IP relatives aux réseaux sous Windows](#)

Ping

PING : Teste la connexion réseau avec une adresse IP distante

```
ping -t [IP ou host]
ping -l 1024 [IP ou host]
```

- L'option -t permet de faire des pings en continu jusqu'à Ctrl-C.
 - Si vous avez précisé l'option -t vous pouvez à tout moment avoir des statistiques sans interrompre les requêtes ping en appuyant sur Ctrl+Attn (aussi nommé Ctrl+Pause)

Cette commande est aussi utile pour générer une charge réseau en spécifiant la taille du paquet avec l'option -l et la taille du paquet en octets.

Tracert

TRACERT : Affiche toutes les adresses IP intermédiaires par lesquelles passe un paquet entre la machine locale et l'adresse IP spécifiée.

```
tracert [@IP ou nom du host]
tracert -d [@IP ou nom du host]
```

Cette commande est utile si la commande ping ne donne pas de réponse, afin d'établir à quel niveau la connexion est défaillante.

IpConfig

IPCONFIG : Affiche ou rafraîchit la configuration réseau TCP/IP

```
ipconfig [/all] [/release [carte]] [/renew [carte]] [/flushdns] [/displaydns]
[/registerdns] [-a] [-a] [-a]
```

Cette commande, exécutée sans option, affiche l'adresse IP en cours, le masque réseau ainsi que la passerelle par défaut au niveau des interfaces réseau connues sur la machine locale.

- /all: Affiche toute la configuration réseau, y compris les serveurs DNS, WINS, bail DHCP, etc ...
- /renew [carte]: Renouvelle la configuration DHCP de toutes les cartes (si aucune carte n'est spécifiée) ou d'une carte spécifique avec le paramètre carte. Le nom de carte est celui qui apparaît avec ipconfig sans paramètre.
- /release [carte]: Envoie un message DHCPRELEASE au serveur DHCP pour libérer la configuration DHCP actuelle et annuler la configuration d'adresse IP de toutes les cartes (si aucune carte n'est spécifiée) ou d'une carte spécifique avec paramètre carte. Ce paramètre désactive TCP/IP pour les cartes configurées de manière à obtenir automatiquement une adresse IP.
- /flushdns: Vide et réinitialise le cache de résolution du client DNS. Cette option est utile pour exclure les entrées de cache négatives ainsi que toutes les autres entrées ajoutées de façon dynamique.
- /displaydns: Affiche le cache de résolution du client DNS, qui inclut les entrées préchargées à partir du fichier des hôtes locaux ainsi que tous les enregistrements de ressources récemment obtenus pour les requêtes de noms résolues par l'ordinateur. Le service Client DNS utilise ces informations pour résoudre rapidement les noms fréquemment sollicités, avant d'interroger ses serveurs DNS configurés.
- /registerdns: Actualise tous les baux DHCP et réinscrit les noms DNS.

NetStat

NETSTAT : Affiche l'état de la pile TCP/IP sur la machine locale

```
NETSTAT [-a] [-e] [-n] [-s] [-p proto] [-r] [intervalle]
```

- -a Affiche toutes les connexions et ports d'écoute (Les connexions côté serveur sont normalement inhibées).
- -e Affiche les statistiques Ethernet. Peut être combinée avec l'option -s.
- -n Affiche les adresses et les numéros de port sous forme numérique.
- -p proto Affiche les connexions pour le protocole spécifié par proto; proto peut être TCP ou UDP. Utilisé avec l'option -s pour afficher des statistiques par protocole, proto peut être TCP, UDP, ou IP.
- -r Affiche le contenu de la table de routage.
- -s Affiche les statistiques par protocole. Par défaut, des statistiques sur TCP, UDP et IP sont visualisées; l'option -p peut être utilisée pour spécifier un sous-ensemble du défaut.
- intervalle: Réaffiche les statistiques sélectionnées, avec une pause de "intervalle" secondes entre chaque affichage. Appuyez sur Ctrl+C pour arrêter l'affichage des statistiques.

- -abnov Affiche les processus qui utilisent la connexion internet (Adresse IP local, port, adress IP distante et le PID du processus qui utilise la connexion ainsi que son nom).

Route

ROUTE : Affiche ou modifie la table de routage

ROUTE [-f] [commande [destination] [MASK masque réseau] [passerelle]

- -f Efface les tables de routage de toutes les entrées de passerelles. Utilisé conjointement à une des commandes, les tables sont effacées avant l'exécution de la commande.
- -p Rend rémanente l'entrée dans la table après le reboot de la machine
- commande Spécifie une des quatre commandes :
 - DELETE: Efface un itinéraire.
 - PRINT: Affiche un itinéraire.
 - ADD: Ajoute un itinéraire.
 - CHANGE: Modifie un itinéraire existant.
- destination: Spécifie l'hôte.
- MASK: Si le mot clé MASK est présent, le paramètre suivant est interprété comme le paramètre masque réseau.
- masque réseau: Fourni, il spécifie la valeur de masque de sous-réseau à associer à cette entrée d'itinéraire. Non spécifié, il prend la valeur par défaut 255.255.255.255.
- passerelle: Spécifie la passerelle.
- METRIC: Spécifie le coût métrique pour la destination

Arp

ARP : Résolution des adresses IP en adresses MAC. Affiche et modifie les tables de traduction des adresses IP en adresses physiques utilisées par le protocole de résolution d'adresses ARP.

ARP -s adr_inet adr_eth [adr_if]

ARP -d adr_inet [adr_if]

ARP -a [adr_inet] [-N adr_if]

- -a Affiche les entrées ARP actives en interrogeant le protocole de données actif. Si adr_inet est spécifié, seules les adresses IP et physiques de l'ordinateur spécifié sont affichées. Si plus d'une interface réseau utilise ARP, les entrées de chaque table ARP sont affichées.
- -g Identique à -a.
- adr_inet Spécifie une adresse internet.
- -N adr_if Affiche les entrées ARP pour l'interface réseau spécifiée par adr_if.
- -d Efface l'hôte spécifié par adr_inet.

- -s Ajoute l'hôte et associe l'adresse Internet adr_inet avec l'adresse physique adr_eth. L'adresse physique est donnée sous forme de 6 octets hexadécimaux séparés par des tirets. L'entrée est permanente.
- adr_eth Spécifie une adresse physique.
- adr_if Précisée, elle spécifie l'adresse Internet de l'interface dont la table de traduction des adresses devrait être modifiée. Non précisée, la première interface applicable sera utilisée.

NbtStat

NBTSTAT : Mise à jour du cache du fichier Lmhosts. Affiche les statistiques du protocole et les connexions TCP/IP actuelles utilisant NBT (NetBIOS sur TCP/IP).

```
NBTSTAT [-a Nom Distant] [-A adresse IP] [-c] [-n] [-r] [-R] [-s] [S]
[intervalle]
```

- -a (état carte) Liste la table de noms de la machine distante (nom connu).
- -A (état carte) Liste la table de noms de la machine distante (adresse IP).
- -c (cache) Liste le cache de noms distant y compris les adresses IP.
- -n (noms) Liste les noms NetBIOS locaux.
- -r (résolus) Liste les noms résolus par diffusion et via WINS.
- -R (Recharge) Purge et recharge la table du cache de noms distante.
- -S (Sessions) Liste la table de sessions avec les adresses destination IP.
- -s (sessions) Liste la table de sessions convertissant les adresses de destination IP en noms d'hôtes via le fichier hôtes.

Un Exemple :

```
nbtstat -A @IP
```

Cette commande renvoie le nom NetBIOS, nom du système, les utilisateurs connectés ...de la machine distante.

Telnet

TELNET

```
telnet <IP ou host>
telnet <IP ou host> <port TCP>
```

La commande telnet permet d'accéder en mode Terminal (Ecran passif) à un host distant. Elle permet également de vérifier si un service quelconque TCP tourne sur un serveur distant en spécifiant après l'adresse IP le numéro de port TCP. C'est ainsi que l'on peut tester si le service

SMTP, par exemple, tourne sur un serveur Microsoft Exchange en utilisant l'adresse IP du connecteur SMTP et puis 25 comme numéro de port. Les ports les plus courants sont :

- ftp (21),
- telnet (23),
- smtp (25),
- www (80),
- kerberos (88),
- pop3 (110),
- nntp (119)
- et nbt (137-139).

Hostname

HOSTNAME : Affiche le nom de la machine

Ftp

FTP: Client de téléchargement de fichiers

```
ftp -s:<file>
```

- -s cette option permet de lancer des FTP en mode batch : spécifie un fichier textuel contenant les commandes FTP.

NsLookUp

NsLookUp: envoie des requêtes DNS sur un serveur DNS au choix

```
nslookup [domaine] [serveur dns]
```

La commande NsLookUp permet d'envoyer des requêtes DNS à un serveur. Par défaut, si vous ne mettez pas le serveur DNS, la commande utilisera celui qui est configuré pour votre interface réseau (celui que vous utilisez pour naviguer sur internet, par exemple) mais vous pouvez forcer l'utilisation d'un autre serveur.

Par exemple, pour demander au serveur DNS 10.0.0.3 l'adresse IP correspondante à l'adresse www.commentcamarche.net:

```
nslookup www.commentcamarche.net 10.0.0.3
```

- Si vous ne précisez aucun paramètre pour nslookup, un shell s'ouvrira en attente de requêtes de votre part.

NetSh

NetSh: configure le réseau sous Windows

Netsh (pour network [shell](#) en anglais) est un [logiciel utilitaire](#) qui présente une [interface utilisateur](#) en [ligne de commande](#) de type [Win32](#) pour la gamme des systèmes d'exploitation [Windows NT](#) (NT, 2000, XP, 2003 Serveur, Vista, etc.) à partir de Windows 2000. Il permet la configuration du [réseau](#), localement ou à distance.

Une utilisation classique de netsh est la réinitialisation de la pile [TCP/IP](#) à ses paramètres d'origine (Sous [Windows 98](#), cette opération nécessitait la réinstallation de l'adaptateur TCP/IP). Dans ce mode, il faut fournir à la commande un fichier journal (log). Celui-ci sera rempli avec les valeurs affectées par netsh.

Netsh permet aussi (entre autres) de changer l'[adresse IP](#) de la machine.

Exemples d'utilisation

Réinitialisation de la pile TCP/IP :

```
netsh interface ip reset C:\resetlog.txt
```

Adresse IP statique :

```
netsh interface ip set address "Connexion au réseau local" static  
123.123.123.123 255.255.255.0
```

Adresse IP dynamique :

```
netsh interface ip set address "Connexion au réseau local" dhcp
```

Netsh et [IPv6](#)

Netsh peut également être utilisé pour lire des informations provenant de la pile [IPv6](#), cette commande est plus facile à utiliser que la commande IPv6.exe et fournit quasiment le même niveau d'information.

Pour afficher votre adresse IPv6 en utilisant netsh :

```
netsh interface ipv6 show address
```

Exemples d'utilisation de NETSH (anglais)

Netsh is a command-line scripting utility that allows you to, either locally or remotely, display or modify the network configuration of a computer that is currently running. **Netsh** also provides a scripting feature that allows you to run a group of commands in batch mode against a specified computer. **Netsh** can also save a configuration script in a text file for archival purposes or to help you configure other servers.

Netsh contexts

Netsh interacts with other operating system components using dynamic-link library (DLL) files. Each Netsh helper DLL provides an extensive set of features called a context, which is a group of commands specific to a networking component. These contexts extend the functionality of **netsh** by providing configuration and monitoring support for one or more services, utilities, or protocols. For example, Dhcpmon.dll provides **netsh** the context and set of commands necessary to configure and manage DHCP servers.

To run a **netsh** command, you must start **netsh** from the Cmd.exe prompt and change to the context that contains the command you want to use. The contexts that are available to you depend on which networking components you have installed. For example, if you type **dhcp** at the Netsh command prompt, you change to the DHCP context, but if you do not have DHCP installed the following message appears:

The following command was not found: dhcp.

For more information about Netsh contexts, see the following topics:

- [Netsh commands for AAAA](#)
- [Netsh commands for DHCP](#)
- [Netsh diagnostic \(diag\) commands](#)
- [Netsh commands for Interface IP](#)
- [Netsh commands for RAS](#)
- [Netsh commands for Routing](#)
- [Netsh commands for WINS](#)

[Top of page](#)

Using multiple contexts

A context can exist within a context. For example, within the Routing context, you can change to the IP and IPX subcontexts.

To display a list of commands and subcontexts that you can use within a context, at the **netsh** prompt, type the context name, and then type either **/?** or **help**. For example, to display a list of

subcontexts and commands that you can use in the Routing context, at the **netsh** prompt (that is, netsh>), type either of the following:

routing /?

routing help

To perform tasks in another context without changing from your current context, type the context path of the command you want to use at the **netsh** prompt. For example, to add the Local Area Connection interface in the IGMP context without changing to the IGMP context, at the **netsh** prompt, type:

routing ip igmp add interface "Local Area Connection" startupqueryinterval=21

[Top of page](#)

Running Netsh commands from the Cmd.exe command prompt

When you run Netsh from the Cmd.exe command prompt, **netsh** uses the following syntax. To run these Netsh commands on a remote Windows 2000 Server, you must first use Remote Desktop Connection to connect to a Windows 2000 Server that is running Terminal Server. There might be functional differences between Netsh context commands on Windows 2000 and Windows XP.

To view the command syntax, click the following command:

netsh

Netsh is a command-line scripting utility that allows you to, either locally or remotely, display or modify the network configuration of a currently running computer. Used without parameters, **netsh** opens the Netsh.exe command prompt (that is, netsh>).

Syntax

netsh [-a *AliasFile*] [-c *Context*] [-r *RemoteComputer*] [{*NetshCommand*|-f *ScriptFile*}]

Parameters

-a : Returns you to the **netsh** prompt after running *AliasFile*.

AliasFile : Specifies the name of the text file that contains one or more **netsh** commands.

-c : Changes to the specified **netsh** context.

Context : Specifies the **netsh** context. The following table lists the available **netsh** contexts.

Context	Description
AAAA	Shows and sets the configuration of the authentication, authorization, accounting, and auditing (AAAA) database used by the Internet Authentication Service (IAS) and the Routing and Remote Access service.
DHCP	Administers DHCP servers and provides an equivalent alternative to console-based management.
Diag	Administers and troubleshoots operating system and network service parameters.
Interface	Configures the TCP/IP protocol (including addresses, default gateways, DNS servers, and WINS servers) and displays configuration and statistical information.
RAS	Administers remote access servers.
Routing	Administers Routing servers.
WINS	Administers WINS servers.

-r : Configures a remote computer.

RemoteComputer : Specifies the remote computer to configure.

NetshCommand : Specifies the **netsh** command that you want to run.

-f : Exits Netsh.exe after running the script.

ScriptFile : Specifies the script that you want to run.

/? : Displays help at the command prompt.

Remarks

- Using **-r**

If you specify **-r** followed by another command, **netsh** executes the command on the remote computer and then returns to the Cmd.exe command prompt. If you specify **-r** without another command, **netsh** opens in remote mode. The process is similar to using **set machine** at the Netsh command prompt. When you use **-r**, you set the target computer for the current instance of **netsh** only. After you exit and reenter **netsh**, the target computer is reset as the local computer. You can run **netsh** commands on a remote computer by specifying a computer name stored in WINS, a UNC name, an internet name to be resolved by the DNS server, or a numerical IP address.

[Top of page](#)

Running Netsh commands from the Netsh.exe command prompt

Netsh uses the following standard commands in all contexts that you can run from a Netsh.exe command prompt (that is, netsh>). To run these Netsh commands on a remote Windows 2000 Server, you must first use Remote Desktop Connection to connect to a

Windows 2000 Server that is running Terminal Server. There might be functional differences between Netsh context commands on Windows 2000 and Windows XP.

To view the command syntax, click a command:

..

Moves to the context that is one level up.

Syntax

..

Parameters

/? : Displays help at the command prompt.

abort

Discards any changes made in offline mode. **Abort** has no effect in online mode.

Syntax

abort

Parameters

/? : Displays help at the command prompt.

add helper

Installs the helper DLL in **netsh**.

Syntax

add helper *DLLName*

Parameters

DLLName : Required. Specifies the name of the helper DLL you want to install.

/? : Displays help at the command prompt.

alias

Adds an alias that consists of a user-defined character string, which **netsh** treats as equivalent to another character string. Used without parameters, **alias** displays all available aliases.

Syntax

alias [*AliasName*] [*string1* [*string2* ...]]

Parameters

alias [*AliasName*] : Displays the specified alias.

alias [*AliasName*][*string1* [*string2* ...]] : Sets *AliasName* to the specified string(s).

/? : Displays help at the command prompt.

Examples

The following **netsh** sample script sets two **netsh** aliases, Shaddr and Shp, and then leaves the Netsh command prompt in the Interface IP context:

```
alias shaddr show interface ip addr
alias shp show helpers
interface ip
```

If you type **shaddr** at the Netsh command prompt, Netsh.exe interprets this as the command **show interface ip addr**. If you type **shp** at the Netsh command prompt, Netsh.exe interprets this as the command **show helpers**.

bye

Exits Netsh.exe.

Syntax

bye

Parameters

/? : Displays help at the command prompt.

commit

Commits any changes made in the offline mode to the router. **Commit** has no effect in online mode.

Syntax

commit

Parameters

/? : Displays help at the command prompt.

delete helper

Removes the helper DLL from **netsh**.

Syntax

delete helper *DLLName*

Parameters

DLLName : Required. Specifies the name of the helper DLL you want to uninstall.

/? : Displays help at the command prompt.

dump

Creates a script that contains the current configuration. If you save this script to a file, you can use the file to restore configuration settings that have been changed. Used without parameters, **dump** displays all of the **netsh** context configurations.

Syntax

dump [*FileName*]

Parameters

[**FileName**] : Specifies the name of the file to which you want to redirect output.

/? : Displays help at the command prompt.

exec

Loads a script file and runs commands from it.

Syntax

exec *ScriptFile*

Parameters

ScriptFile : Required. Specifies the name of the script that you want to load and run.

Remarks

- The *ScriptFile* can run on one or more computers.

exit

Exits Netsh.exe.

Syntax

exit

Parameters

/? : Displays help at the command prompt.

help

Displays help.

Syntax

{/?|?|help|h}

Parameters

none

offline

Sets the current mode to offline.

Syntax

offline

Parameters

/? : Displays help at the command prompt.

Remarks

- Changes that you make in this mode are saved, but you need to run the **commit** or **online** command to set the changes in the router.
- When you switch from offline mode to online mode, changes that you made in offline mode are reflected in the configuration that is currently running.
- Changes that you make in online mode are immediately reflected in the configuration that is currently running.

online

Sets the current mode to online.

Syntax

online

Parameters

/? : Displays help at the command prompt.

Remarks

- Changes that you make in online mode are immediately reflected in the configuration that is currently running.
- When you switch from offline mode to online mode, changes that you made in offline mode are reflected in the configuration that is currently running.

popd

Restores a context from the stack.

Syntax

popd

Parameters

/? : Displays help at the command prompt.

Remarks

- Used in conjunction with **pushd**, **popd** enables you to change the context, run the command in the new context, and then resume the prior context.

Examples

The following sample script changes a context from the root context to the **interface ip** context, adds a static IP route, and then returns to the root context:

```
netsh>
```

```
pushd
```

```
netsh>
```

```
interface ip
```

```
netsh interface ip>
```

```
set address local static 10.0.0.9 255.0.0.0 10.0.0.1 1
```

```
netsh interface ip>
```

```
popd
```

```
netsh>
```

```
pushd
```

Saves the current context on a first-in-last-out (FILO) stack.

Syntax

```
pushd
```

Parameters

/? : Displays help at the command prompt.

Remarks

- Used in conjunction with **popd**, **pushd** enables you to change the context, run the command in the new context, and then resume the prior context.

```
quit
```

Exits Netsh.exe.

Syntax

```
quit
```

Parameters

/? : Displays help at the command prompt.

set file

Copies the Command Prompt window output to a file.

Syntax

set file { **open** *FileName* | **append** *FileName* | **close** }

Parameters

open *FileName* : Sends the Command Prompt window output to the specified file.

append *FileName* : Appends the Command Prompt window output to the specified existing file.

close : Stops sending output and closes a file.

/? : Displays help at the command prompt.

Remarks

- Using *FileName*

If the specified *FileName* does not currently exist, **netsh** creates a new file with that name. If the specified *FileName* does currently exist, **netsh** overwrites the existing data.

Examples

To create a new log file called Session.log and copy all succeeding **netsh** input and output to Session.log, type:

```
set file open c:\session.log
```

```
set machine
```

Sets the current computer on which to perform configuration tasks. Used without parameters, **set machine** sets the local computer.

Syntax

```
set machine [[ComputerName=]string]
```


Parameters

ComputerName : Specifies the name of the computer on which to perform configuration tasks.

string : Specifies the location of the remote computer.

/? : Displays help at the command prompt.

Remarks

- Performing configuration tasks on multiple computers

You can run commands on multiple computers from a single script. You can use **set machine** in a script to specify a destination computer (for example, ComputerA), and then run the commands that follow **set machine** on this computer (that is, ComputerA). You can then use **set machine** to specify another destination computer (for example, ComputerB), and then run commands on this computer (that is, ComputerB).

set mode

Sets the current mode to online or offline.

Syntax

set mode {online|offline}

Parameters

online : Sets the current mode to online.

offline : Sets the current mode to offline.

/? : Displays help at the command prompt.

show

Displays alias, helper, and mode information.

Syntax

show {alias|helper|mode}

Parameters

alias : Lists all defined aliases.

helper : Lists all top-level helpers.

mode : Displays the current mode.

/? : Displays help at the command prompt.

unalias

Deletes the specified alias.

Syntax

unalias *AliasName*

Parameters

AliasName : Required. Specifies the name of the alias.

/? : Displays help at the command prompt.

[Top of page](#)

Formatting legend

	Format	Meaning
	<i>Italic</i>	Information that the user must supply
	Bold	Elements that the user must type exactly as shown
	Ellipsis (...)	Parameter that can be repeated several times in a command line
	Between brackets ([])	Optional items
	Between braces ({}); choices separated by pipe (). Example: {even odd}	Set of choices from which the user must choose only one
	Courier font	Code or program output