Object Filler and Object Dumper Tutorial

A guide for beginners on the art of bulk objects creation and exporting with Check Point solutions

Martín Humberto Hoz Salvador, CISA, CISSP

martinhoz at gmail dot com
mhoz at mexico dot com

Created on November 2005

Revision 20061220

Table of Contents

1. Introduction

This document has the objective of teaching you the basics on Object Filler and Object Dumper, as well on some of the internals in Objects Database Management with Check Point products, so you can benefit from both tools (Object Filler and Object Dumper) in your current Check Point environment. It focuses in the more common (and sometimes weird) situations for Check Point administrators, and details the operations performed in a step-by-step basis, so you can follow them as exercises. It should take you (with the proper software installed) between 3 to 5 hours to read all of it and follow the exercises. But it's worth the time…

This document is available as part of the Object Filler's package. This document is also available together with all the files mentioned through the whole document, as a separate package. Please refer to the websites mentioned below so you can access to all the files for your practice if you like. This way you can compare the results you got with the results you see printed in

The Tutorial is supposed to be read-through, even though it can be helpful as well as a (not so quick) reference.

The idea is: If you follow the examples, they will show you how the tools work and thus you will learn how they can be helpful for you. Even though the document seems to be long, it is actually not. Thing is that lots of screen shots were taken to make the document more clear and that's why it looks overwhelming. Besides, I promise you that most of the reading won't be boring and will pay off really well later… and I'm known to be a man of my word ;-)

Warning! Yep, the style of writing I've to admit is not formal and a bit irreverent, but still respectful to the reader. Why this? Well, I'm writing the first 40 pages of this document on vacation time because I was getting bored of being on vacation ☺. I needed some radiation from the screen ;-), so there you go… - the good thing is that (I believe) this makes the documentation funnier and easier to read. Let me know what you think please…

This document is NOT a replacement for the Object Filler and Object Dumper User's Manual, but instead a guide to lead you to learn how to use the tools faster than reading all the documentation. It is suggested for you to read the User's Manual as well, and have it with you as a reference while reading this document.

It is good to remember that these tools (i.e. Object Filler and Object Dumper) are currently NOT official nor supported, and suggested you take every possible caution before using the tools in a production environment.

You may find more information on Object Filler and Object Dumper here:
http://ofiller.chatscope.com (main download site with FAQs, bug tracker, beta downloads and other nice resources)
http://www.lindercentral.com/ofiller/ - Always with the latest stable version

http://www.cpug.org/check_point_resources.htm (http://www.cpug.org) - Always with the latest stable version

This document assumes that you already have some experience and know how to operate a Check Point VPN-1 Power installation, that you know the architectural concepts of SmartCenter, SmartConsole and Gateway, as well as the basics on rules and objects management. This document assumes as well that you know how to troubleshoot basic issues that may arise with your Check Point installation.

You need to know the basics on Operating System and network engineering to understand firewall technology as well.

This document is NOT a tutorial on firewall management, firewall technology nor a Check Point products introduction.

Finally, if you have Provider-1, then you should know that the procedures do work with Provider-1 as well. However, before doing anything you should set your environment properly using *mdsenv*, so the environment variables to the CMA you want to work with. There is as well a document focused only on how Object Filler and Object Dumper work with Provider-1. You may want to take a look at it if that is your case - If you don't know what Provider-1 or a CMA is, don't worry: the document is still useful for you the way it is… ;-)

Before I'm done here, I want to thank Brad Molles for reviewing this material very thoroughly and correcting several mistakes I had with it initially… (as an excuse, I'm not a native English speaker, so Brad had to deal with that :-P) - Thanks a lot for your time and effort Brad!

1.1 Typographic convention

The regular text of the document will be written in regular Times font..

Text seen in the screen or output shown as result of some command, will be shown on Courier font, like this.

Text that the user has to type-in and is normally shown to the screen will be on Courier bold, as this example

Text that the user has to type-in but is not shown to the screen will be printed in Courier bold and italic font, as in this example

Whenever a new term is introduced, some specific name is used or there is a buzzword, it will be printed in italic format.

Whenever it is needed to highlight something, it will be underlined.

2. Disclaimer

Warning! This is a boring part. But it has to be here… 😐

Object Filler and Object Dumper tools are not officially supported. This means, even though they work, nobody has the obligation of helping you, and if you call Check Point Support they won't be able to assist you. Please read the Object Filler and Object Dumper documentation before you proceed.

Even though the procedures described here have been tested by the author of this document and some others, they are not guaranteed to be error-free. The only responsible entity for the results of using the procedures and techniques described on this document, is the person implementing them and the person that approves the use of this on his infrastructure.

Check Point Software Technologies, Check Point employees and affiliates, and/or the document's author; are not liable nor responsible in any way for any good or bad thing resulting from any direct or indirect use, abuse or misuse (known or unknown) of the contents of this document.

So, in other words, if you get promoted or fired because you used this document somehow, that's always your fault!

All the brands, trademarks, copyrighted material, etc. belong to their respective owners. The only thing of which I can claim something is on the Object Filler and Object Dumper names, as I had a hard time thinking on them.

3. Environment Description

Okay, now to the details of how it was possible to do all what is described here.

The software version used on the first edition document was Check Point VPN-1 NGX R60 installed on SecurePlatform. On it, both SmartCenter and Gateway were installed in the same Machine (StandAlone Installation). The versions used in some sections added in a later review range from R60A to R62. The machine used for this document was a Virtual Machine using VMWare 5 (http://www.vmware.com). The Virtual Machine had 8 GB of defined (not assigned space) SCSI disk, 2 NICs (one host only, one bridged) and 256 MB of RAM. SmartConsole was installed on a Microsoft Windows XP machine, the same where the Object Filler and Object Dumper were running. See details below:

```
[ngxr62]# ver
This is Check Point SecurePlatform Pro NGX (R60) Build 244
[ngxr62]# fw ver
This is Check Point VPN-1(TM) & FireWall-1(R) NGX (R60) - Build 458
```

Latest version used while adding elements to the document:

```
[ngxr62]# ver
This is Check Point SecurePlatform NGX (R62) Build 031
[ngxr62]# fw ver
This is Check Point VPN-1(TM) & FireWall-1(R) NGX (R62) - Build 120
```

The product was licensed using the Plug and Play license (the one that is automatically put in place when you install the software).

The *expert mode* of SecurePlatform was the working environment and it is assumed that you already provided and changed the password for both restricted and expert mode. This is needed because you need to interact with files and do operations that the restricted *cpshell*, which SecurePlatform puts to all the defined users by default, normally doesn't allow. If you want more information, please refer to the SecurePlatform/SecurePlatform Pro NGX R62 User's Guide.

```
Last login: Mon Nov 27 20:56:38 2006 from 10.20.30.1

? for list of commands
sysconfig for system and products configuration

[ngxr62]# expert
Enter expert password: secret

You are in expert mode now.

[Expert@ngxr62]#
```

The tools version used was Object Dumper and Object Filler 2.4 running on Microsoft Windows XP Professional Service Pack 2 with 1 GB of RAM, though the tools should run fine even in machines with 128 MB of RAM. Not even 10 MB of hard disk space was needed for the tools themselves and the files created here.

```
D:\Stuff\OFiller\v2.4>ver

Microsoft Windows XP [Version 5.1.2600]

D:\Stuff\OFiller\v2.4>ofiller -V
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

*******************************************************************************
This is Object Filler Version v2.4
This is Object Filler Build Number Thu_211206_1229
*******************************************************************************

===============================================================================
No valid objects were processed! - Thank you for using Object Filler v2.4!

D:\Stuff\OFiller\v2.4>odumper -V
Unofficial/Unsupported Object Dumper v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

*******************************************************************************
This is Object Dumper Version v2.4
This is Object Dumper Build Number Sat_211206_1659
*******************************************************************************

===============================================================================
No valid objects were processed! - Thank you for using Object Dumper v2.4!
```

For transferring files between machines, an FTP Server (in our case, 3CDaemon Version 2.0 Revision 10 – which is freely available on the Internet) was set up in the Windows machine. SecurePlatform has an FTP client by default, so there was no need of doing something else to transfer files between the 2 machines. If you like, you may use some other file transfer method.

3CDaemon
Version 2.0
Revision 10
Copyright 1999-2000, 3Com Corp.
Questions and comments to dan_gill@3com.com

I would like to make a clear point (once again), that even though all the examples are given around a SmartCenter scenario, it is possible to do all this in a Provider-1 scenario, given the proper environment settings, as stated originally in the introduction.

For what it’s worth, I was drinking lots of Colombian (*Juan Valdés*) and/or Mexican (*Los Portales de Córdoba*) coffee while I wrote this document. Both of them are great!

4. Objects Management Basics

In order to represent a security policy in a firewall, rules have to be created. Such rules enforce if the traffic passing though should be allowed or discarded, according to the security policy.

In Check Point VPN-1 installation (Pro, Express, Power or UTM), rules contain a fixed amount of fields, such as "*source*" (where in the network the communication begins), "*destination*" (the communication's target) or "*service*" (what ports are allowed or prohibited by the rule).

To fill-up such fields, there are *objects* that represent entities in the network that have to be created first. For example, you can define a Node object of type Host to represent a Server or a Workstation in your network. You can define a Network object to represent a certain subnet where some specific users are located. You can define a new TCP Service object to represent that through port TCP/85 there's some traffic that should pass in order for some communication in your network to work.

4.1 Objects storage

When you define objects in SmartDashboard, they are stored in the $FWDIR/conf/objects_5_0.C file. This file is a plain ASCII Text file, but has some special formatting.

```
[Expert@ngxr62]# cd $FWDIR/conf
[Expert@ngxr62]# pwd
/opt/CPsuite-R62/fw1/conf
[Expert@ngxr62]# ls -la objects_5_0.C
-rw-rw----    1 root     root       653878 Nov 21 21:58 objects_5_0.C
[Expert@ngxr62]# file objects_5_0.C
objects_5_0.C: ASCII text
[Expert@ngxr62]#
```

Similarly, policies (which are made of rules) are stored in other plain ASCII Text files.

You may see that for Security Policies are stored individually on files with .W extension. Whenever those policies are compiled, a .pf file is generated containing all the INSPECT code statements. For example, if you have a policy called *MyDemo*, those 2 files are created.

```
[Expert@ngxr62]# pwd
/opt/CPsuite-R60/fw1/conf
[Expert@ngxr62]# ls -la MyDemo1.*
-rw-rw----    1 root     root         1898 Nov 13 12:36 MyDemo1.W
-rw-rw----    1 root     root       127855 Nov 13 12:36 MyDemo1.pf
[Expert@ngxr62]# file MyDemo1.*
MyDemo1.W:  ASCII text
MyDemo1.pf: C++ program text
```

Additionally, all the policies are stored all together in a file called rulebases_5_0.C, which is stored in the same directory.

```
[Expert@ngxr62]# pwd
/opt/CPsuite-R60/fw1/conf
```

```
[Expert@ngxr62]# ls -la rulebases_5_0.fws
-rw-rw----    1 root     root         5350 Nov 13 12:36 rulebases_5_0.fws
[Expert@ngxr62]# file rulebases_5_0.fws
rulebases_5_0.fws: ASCII text
```

If you want, you may use a text editor or a command like *cat* or *more* to take a look at the content of the files. However, you are NOT supposed to manually edit any of these files, or you risk the integrity of them and thus leaving your configuration unstable.

4.2 DBedit

To modify those files, there are two supported means to do it. One is using the SmartDashboard. Whenever you create or modify an object there, it is automatically added to where it belongs once you save your configuration. The other way is using a command line tool called DBedit. The latest is the method we will use mostly through this document.

If you want to know more on this, please consult the Command Line Interface (CLI) Guide NGX (R62) which comes with the regular Check Point documentation. Also the following entries from SecureKnowledge (http://secureknowledge.checkpoint.com) are useful to understand how DBedit works: skI3301, sk10104

Basically the idea with DBedit is to use commands to create or modify objects properties. To see how it works, let's create a simple host object using DBedit.

a) First let's make sure no SmartConsole clients (such as SmartDashboard) are connected to the SmartCenter Server:

```
[Expert@ngxr62]# cpstat mg

Product Name:  Check Point SmartCenter Server
Major version: 6
Minor version: 0
Build number:  618000021
Is started:    1
Active status: active
Status:        OK

Connected clients
-------------------------------------------------
|Client type   |Administrator|Host |Database lock|
-------------------------------------------------
|SmartDashboard|admin        |mhoz1|true         |
-------------------------------------------------
```

This means that there is actually an administrator named *admin* using SmartDashboard connected from a machine named *mhoz1* and is locking the Objects Database. Let's disconnect it. Now, if you issue *cpstat mg* again, the output should look like this:

```
[Expert@ngxr62]# cpstat mg

Product Name:  Check Point SmartCenter Server
Major version: 6
Minor version: 0
Build number:  618000021
Is started:    1
Active status: active
Status:        OK

Connected clients
----------------------------------------------
|Client type|Administrator|Host|Database lock|
----------------------------------------------
----------------------------------------------
```

This seems to be ok now.

b) Then let’s log in with DBedit. This assumes the SmartCenter is local and the administrator username is *admin*.

```
[Expert@ngxr62]# dbedit -s localhost -u admin
Enter Administrator Password: secret

Please enter a command, -h for help or -q to quit:
dbedit>
```

c) Now let’s type in the commands to create a host object named *myserver1* with IP Address *10.20.30.87* and color *green*. A comment that it was a dbedit example will be added as well.

```
dbedit> create host_plain myserver1
dbedit> modify network_objects myserver1 ipaddr 10.20.30.87
dbedit> modify network_objects myserver1 color green
dbedit> modify network_objects myserver1 comments "This is a dbedit example"
dbedit> update network_objects myserver1
myserver1 updated successfully.
```

d) Let’s add a *Hide NAT* behind the IP address *192.168.20.98* and enforced by the gateway named *ngxr62*. Such gateway object <u>must</u> previously exist.

```
dbedit> modify network_objects myserver1 NAT NAT
dbedit> modify network_objects myserver1 add_adtr_rule true
dbedit> modify network_objects myserver1 NAT:valid_ipaddr 192.168.20.98
dbedit> modify network_objects myserver1 NAT:netobj_adtr_method adtr_hide
dbedit> modify network_objects myserver1 NAT:the_firewalling_obj network_objects
:ngxr62
dbedit> update_all
network_objects::myserver1 Updated Successfully
```

e) We need to see how the new object looks from the database perspective

```
dbedit> print network_objects myserver1

Object Name: myserver1
Object UID: {B67E44EC-7E93-11DB-B33C-000000003C3C}
Class Name: host_plain
Table Name: network_objects
Last Modified by: admin
Last Modified from: ngxr62
Last Modification time: Tue Nov 28 03:52:40 2006
Fields Details
--------------
    DAG: false
    NAT: myserver1 (
        netobj_adtr_method: adtr_hide
        the_firewalling_obj: Name: ngxr62 (Table: network_objects)
        valid_addr_name:
        valid_ipaddr: 192.168.20.98
    )
    SNMP: myserver1 (
        read_community:
        sysContact:
        sysDescr:
        sysLocation:
        sysName:
        write_community:
    )
    VPN: (
        <NULL>
    )
    add_adtr_rule: true
    additional_products: (
        <NULL>
    )
    certificates: (
        (
            <NULL>
        )
    )
    color: green
    comments: This is a dbedit example
    cp_products_installed: false
    edges:
    enforce_gtp_rate_limit: false
    firewall: not-installed
    floodgate: not-installed
    gtp_rate_limit: 2048
    interfaces: (
        (
            <NULL>
        )
    )
    ipaddr: 10.20.30.87
    os_info: (
        <NULL>
    )
    type: host
```

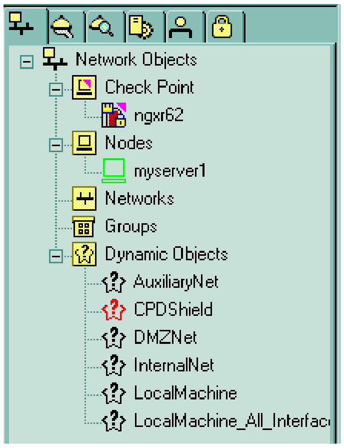
Interesting stuff, isn't it? – All those details are kept in the objects database, but hidden from the regular administrators as usually those details are not needed, unless you are doing some low level stuff…

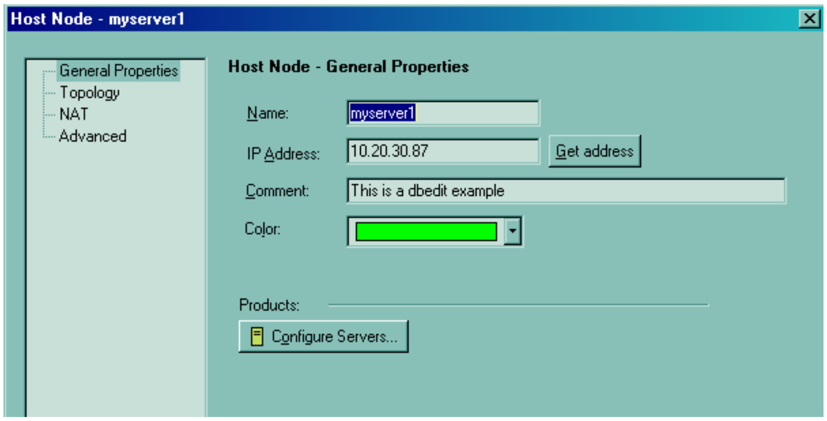
Though we won’t use most of it here, it is good for you to know it, so you can diagnose potential issues with the tools later.

f) Let’s log out from DBedit

```
dbedit> quit
[Expert@ngxr62]#
```

g) Now let’s go to SmartDashboard. Login to your SmartCenter and see how the recently created object looks there:





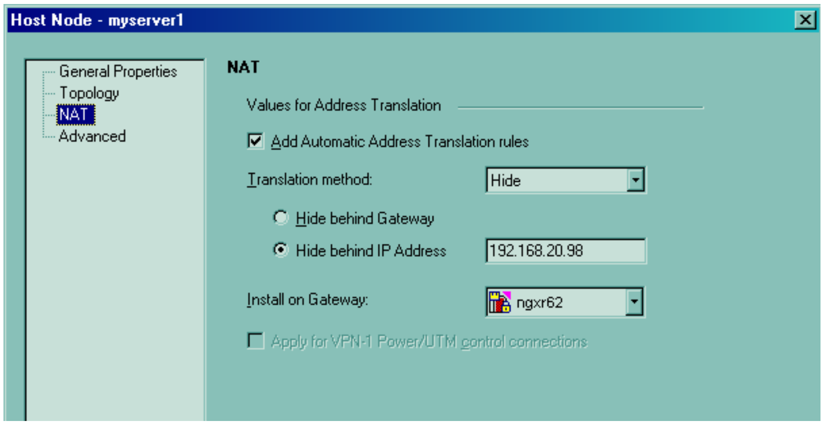
Host Node - General Properties

Name: myserver1

IP Address: 10.20.30.87

Comment: This is a dbedit example

Color: 🟩



NAT

Values for Address Translation

☑ Add Automatic Address Translation rules

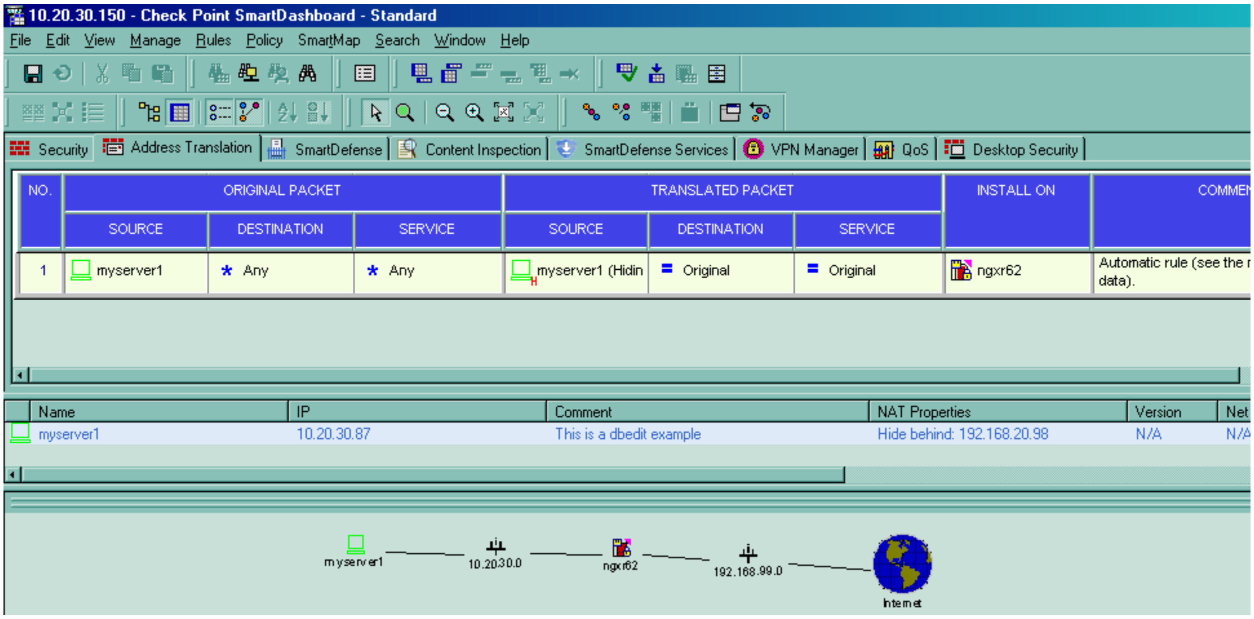
Translation method: Hide

○ Hide behind Gateway

● Hide behind IP Address 192.168.20.98

Install on Gateway: ngxr62

☐ Apply for VPN-1 Power/UTM control connections



NO.	ORIGINAL PACKET			TRANSLATED PACKET			INSTALL ON	COMMEN
	SOURCE	DESTINATION	SERVICE	SOURCE	DESTINATION	SERVICE		
1	myserver1	* Any	* Any	myserver1 (Hidin	Original	Original	ngxr62	Automatic rule (see the r data).

Name	IP	Comment	NAT Properties	Version	Net
myserver1	10.20.30.87	This is a dbedit example	Hide behind: 192.168.20.98	N/A	N/A

As you can see, all is there, just as if you would done it from SmartDashboard…

The difference is that you created it using a command line tool, which seems more complex, and in fact it is more complex because you need to know or remember all the instructions and the differences in the properties for the different kind of objects; but is simpler to script, which has advantages for certain tasks like bulk objects creation.

5. Object Filler

So, how to make things simpler? How to make the creation of DBedit scripts easier without remembering and verifying the syntax all the time?

This is where Object Filler comes handy. Object Filler is a tool that takes as input what objects you want to create, and produces DBedit commands that you can enter directly into your SmartCenter to effectively create such objects.

This approach has several advantages: Instead of having to remember all the DBedit syntax and options, you just provide the information for what you want to do. Since the output is text, you can see what the DBedit commands look like, and modify something if you want before doing the operation into your SmartCenter. Actually, there is an option available so you can preview the output in a very nice way using a SpreadSheet able to interpret CSV (Comma Separated Values) files. Even more, Object Filler has built-in the capability of reading configuration files from Cisco PIX, Cisco Routers, Juniper/NetScreen, Symantec Raptor, Gauntlet and SideWinder firewalls; so you don’t have to bother manually analyzing and then typing the configuration to a brand new powerful Check Point VPN-1 Power firewall…

Looks/Seems/Sounds nice? – Let’s take a look on how you can do all that!

5.1 Creating and importing consecutive objects

One of the most boring parts when installing a new firewall is when you have several consecutive objects, such as all the 254 subnets for a Class B WAN when subnetted to 24 bits subnets, or all the 254 hosts within a Class C network. Let’s see how you can, once you know the syntax well, build all those objects in less than 5 minutes. Guaranteed!

5.1.1 Creating consecutive networks

Let’s say you have the 10.10.0.0/16 network and you need to build for a customer all the around 250 subnets that result of subnetting it to 24 bits. This means you would have to create a 10.10.0.0/24 object, then a 10.10.1.0/24 object, later a 10.10.2.0/24 object and so on… boring and tiring…

Object Filler can help, if you follow this procedure:

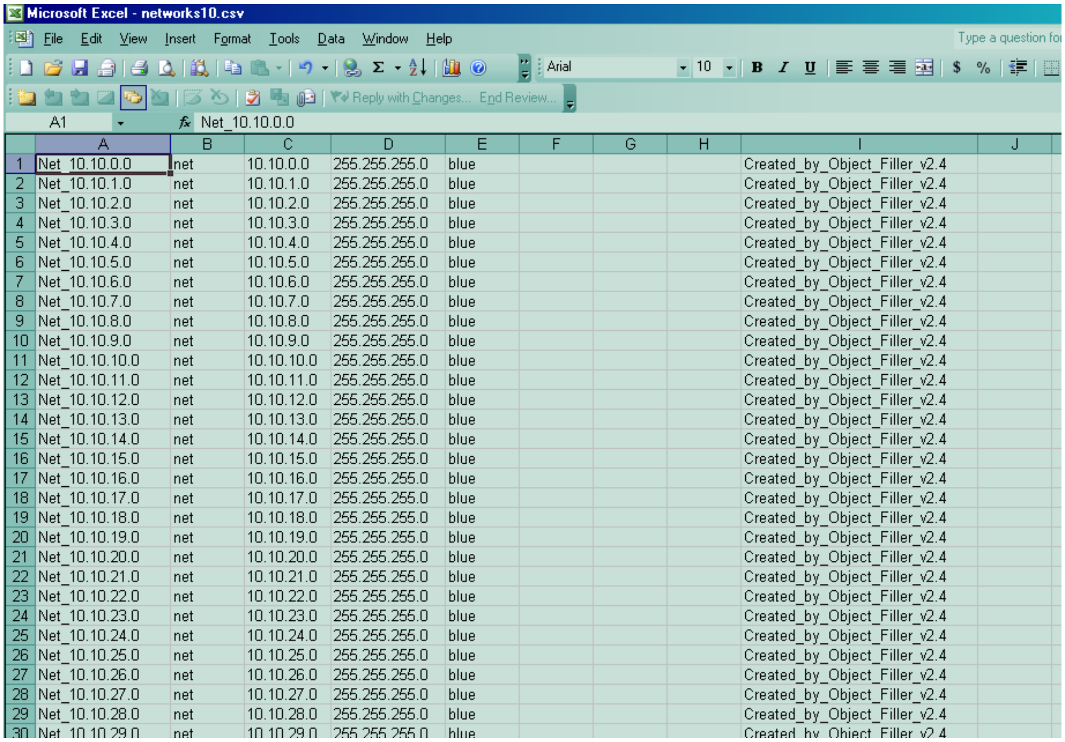
a) You know that your initial IP would be *10.10.0.0*, your final IP would be *10.10.255.255* and your netmask is *24* bits. You want just *networks*. Let’s say you want the objects to be of color blue and you want to preview first that everything is ok using a Spreadsheet program like Microsoft Excel. You may feed Object Filler with this information:

```
D:\Stuff\OFiller\v2.4> ofiller -s 10.10.0.0 -d 10.10.255.255 -m 24 -c blue -t net
s -a networks10.csv
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
```

```
==============================================================================
....................................................
==============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 65306 possible objects and/or rules.
Found 256 total valid (or successfully processed) objects/rules.
------------------------------------------------------------------------------
Total successfully processed Networks = 256
------------------------------------------------------------------------------
Please review that all CSV output information was written correctly.
Please remember CSV information is for reviewing only. For importing into
SmartCenter you need to use DBedit mode (-o switch).
==============================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

Great! – Now what we told Object Filler was to begin from 10.10.0.0 (`-s 10.10.0.0`) up to 10.10.255.255 (`-d 10.10.255.255`) with a 24 bits netmask (`-m 24`). The color was set to blue (`-c blue`) the type of objects to be created is networks (`-t nets`) and finally the output was directed to the `networks10.csv` file. Notice we used the switch "`–a`" to direct the output, as we want to preview with a spreadsheet (Microsoft Excel in this case) what the result was.

Let's do it. Let's open the file with Microsoft Excel and see how it looks.


Microsoft Excel - networks10.csv
File Edit View Insert Format Tools Data Window Help
Type a question fo
Arial
10
B I U
$ %
Reply with Changes... End Review...

A1 fx Net_10.10.0.0

	A	B	C	D	E	F	G	H	I	J
1	Net_10.10.0.0	net	10.10.0.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
2	Net_10.10.1.0	net	10.10.1.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
3	Net_10.10.2.0	net	10.10.2.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
4	Net_10.10.3.0	net	10.10.3.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
5	Net_10.10.4.0	net	10.10.4.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
6	Net_10.10.5.0	net	10.10.5.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
7	Net_10.10.6.0	net	10.10.6.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
8	Net_10.10.7.0	net	10.10.7.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
9	Net_10.10.8.0	net	10.10.8.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
10	Net_10.10.9.0	net	10.10.9.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
11	Net_10.10.10.0	net	10.10.10.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
12	Net_10.10.11.0	net	10.10.11.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
13	Net_10.10.12.0	net	10.10.12.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
14	Net_10.10.13.0	net	10.10.13.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
15	Net_10.10.14.0	net	10.10.14.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
16	Net_10.10.15.0	net	10.10.15.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
17	Net_10.10.16.0	net	10.10.16.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
18	Net_10.10.17.0	net	10.10.17.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
19	Net_10.10.18.0	net	10.10.18.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
20	Net_10.10.19.0	net	10.10.19.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
21	Net_10.10.20.0	net	10.10.20.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
22	Net_10.10.21.0	net	10.10.21.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
23	Net_10.10.22.0	net	10.10.22.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
24	Net_10.10.23.0	net	10.10.23.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
25	Net_10.10.24.0	net	10.10.24.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
26	Net_10.10.25.0	net	10.10.25.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
27	Net_10.10.26.0	net	10.10.26.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
28	Net_10.10.27.0	net	10.10.27.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
29	Net_10.10.28.0	net	10.10.28.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	
30	Net_10.10.29.0	net	10.10.29.0	255.255.255.0	blue				Created_by_Object_Filler_v2.4	

The order of the columns shown won't be fully explained here as this is documented in the Users Manual, but let's say the first column is the name (Automatically given), the second is the object type, the third IP address, the fourth is the netmask, the fifth is the color, the sixth is the NATting

IP address (if it is blank, no NAT is assumed), the seventh is the Gateway enforcing such NAT, the eight is the NAT Type (Static or Hide) and the ninth is the comment.

Well it seems to be something like what we need… but wait a minute! – We forgot that may be all those objects are to be NATted with *Hide NAT* behind IP addresses that range from *192.168.100.101* to *192.169.100.110* (which is 10 IP address for all the 255 networks) and protected behind the gateway *ngxr62*.

Let's do it all this with Object Filler.

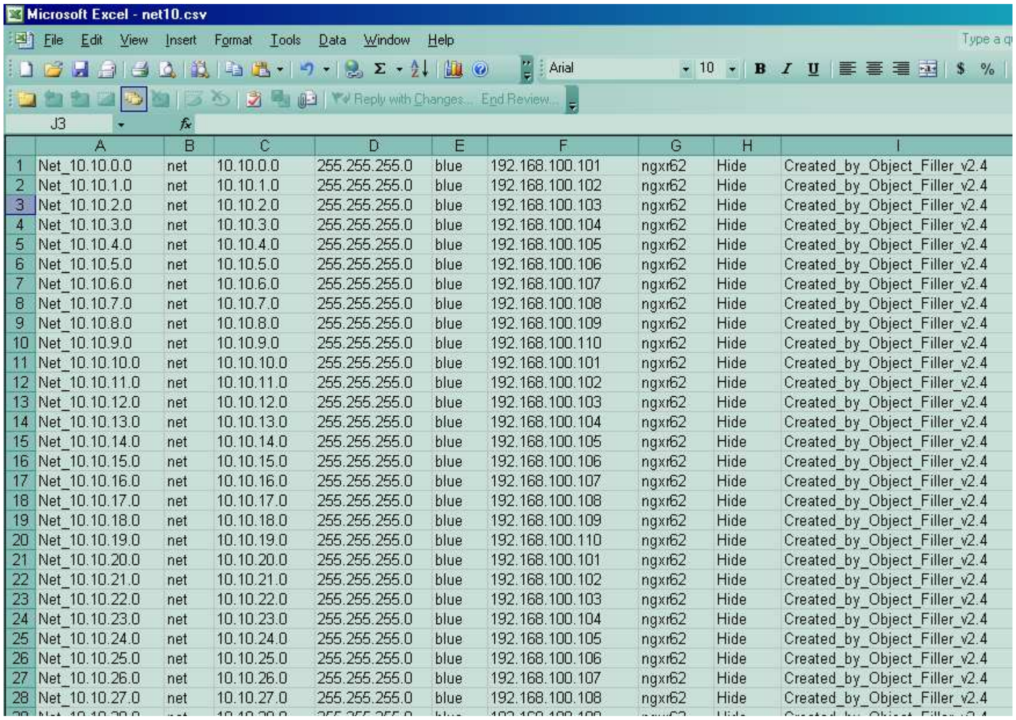
```
D:\Stuff\OFiller\v2.4>ofiller -s 10.10.0.0 -d 10.10.255.25 -m 24 -c blue -t net
-ns 192.168.100.101 -nd 192.168.100.110 -nm 24 -b ngxr62 -a net10.csv
Unofficial/Unsupported Object Filler V2.4  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
.....................................................
===============================================================================
It took 4 seconds of total processing time on QUIET Mode.
Processed 10 NATting IPs, 65306 possible objects and/or rules.
Found 256 total valid (or successfully processed) objects/rules.
Found 256 total NATted records.
-------------------------------------------------------------------------------
Total successfully processed Networks = 256
-------------------------------------------------------------------------------
Please review that all CSV output information was written correctly.
Please remember CSV information is for reviewing only. For importing into
SmartCenter you need to use DBedit mode (-o switch).
===============================================================================
Task done successfully! - Thank you for using Object Filler V2.4!
```

Please notice that the new arguments are related to the Hide NAT: `-ns 192.168.100.101` indicates that there is a NAT range that begins with *192.168.100.101* and ends (`-nd`) with *192.169.100.110*. `–nm 24` is needed because it tells Object Filler when to skip addresses that may not be valid IP addresses when the Hidding IP Addresses belong to different subnets. `–b ngxr62` tells Object Filler to Hide all the created objects behind such network object, which must previously exist and be exactly named (case sensitive) the way it is typed here. Also, the name of the output file (`-a net10.csv`) has to be changed as Object Filler does NOT overwrite files already present in disk, just as a caution so files are not accidentally overwritten.

Let's analyze the output a bit. The dots you see printed over there are activity indicators. This is, while Object Filler is working it prints some dots, so you know is busy doing something and it has not died. This is useful when processing a large number of objects and there are more complex operations that take some time.

The output also gives you some statistics on what the processed stuff is. In this example, it tells you 256 network objects were processed, and it processed 10 NATting IP addreses. It tells you as well that is CSV format and NOT DBedit commands what it was written.

Now let's see how the resulting file looks. Pay attention to the 6th column that shows the NATting IP address:



Microsoft Excel - net10.csv
File Edit View Insert Format Tools Data Window Help
Type a q
Arial
10
B I U
$ %
Reply with Changes... End Review...
J3
fx

	A	B	C	D	E	F	G	H	I
1	Net_10.10.0.0	net	10.10.0.0	255.255.255.0	blue	192.168.100.101	ngxr62	Hide	Created_by_Object_Filler_v2.4
2	Net_10.10.1.0	net	10.10.1.0	255.255.255.0	blue	192.168.100.102	ngxr62	Hide	Created_by_Object_Filler_v2.4
3	Net_10.10.2.0	net	10.10.2.0	255.255.255.0	blue	192.168.100.103	ngxr62	Hide	Created_by_Object_Filler_v2.4
4	Net_10.10.3.0	net	10.10.3.0	255.255.255.0	blue	192.168.100.104	ngxr62	Hide	Created_by_Object_Filler_v2.4
5	Net_10.10.4.0	net	10.10.4.0	255.255.255.0	blue	192.168.100.105	ngxr62	Hide	Created_by_Object_Filler_v2.4
6	Net_10.10.5.0	net	10.10.5.0	255.255.255.0	blue	192.168.100.106	ngxr62	Hide	Created_by_Object_Filler_v2.4
7	Net_10.10.6.0	net	10.10.6.0	255.255.255.0	blue	192.168.100.107	ngxr62	Hide	Created_by_Object_Filler_v2.4
8	Net_10.10.7.0	net	10.10.7.0	255.255.255.0	blue	192.168.100.108	ngxr62	Hide	Created_by_Object_Filler_v2.4
9	Net_10.10.8.0	net	10.10.8.0	255.255.255.0	blue	192.168.100.109	ngxr62	Hide	Created_by_Object_Filler_v2.4
10	Net_10.10.9.0	net	10.10.9.0	255.255.255.0	blue	192.168.100.110	ngxr62	Hide	Created_by_Object_Filler_v2.4
11	Net_10.10.10.0	net	10.10.10.0	255.255.255.0	blue	192.168.100.101	ngxr62	Hide	Created_by_Object_Filler_v2.4
12	Net_10.10.11.0	net	10.10.11.0	255.255.255.0	blue	192.168.100.102	ngxr62	Hide	Created_by_Object_Filler_v2.4
13	Net_10.10.12.0	net	10.10.12.0	255.255.255.0	blue	192.168.100.103	ngxr62	Hide	Created_by_Object_Filler_v2.4
14	Net_10.10.13.0	net	10.10.13.0	255.255.255.0	blue	192.168.100.104	ngxr62	Hide	Created_by_Object_Filler_v2.4
15	Net_10.10.14.0	net	10.10.14.0	255.255.255.0	blue	192.168.100.105	ngxr62	Hide	Created_by_Object_Filler_v2.4
16	Net_10.10.15.0	net	10.10.15.0	255.255.255.0	blue	192.168.100.106	ngxr62	Hide	Created_by_Object_Filler_v2.4
17	Net_10.10.16.0	net	10.10.16.0	255.255.255.0	blue	192.168.100.107	ngxr62	Hide	Created_by_Object_Filler_v2.4
18	Net_10.10.17.0	net	10.10.17.0	255.255.255.0	blue	192.168.100.108	ngxr62	Hide	Created_by_Object_Filler_v2.4
19	Net_10.10.18.0	net	10.10.18.0	255.255.255.0	blue	192.168.100.109	ngxr62	Hide	Created_by_Object_Filler_v2.4
20	Net_10.10.19.0	net	10.10.19.0	255.255.255.0	blue	192.168.100.110	ngxr62	Hide	Created_by_Object_Filler_v2.4
21	Net_10.10.20.0	net	10.10.20.0	255.255.255.0	blue	192.168.100.101	ngxr62	Hide	Created_by_Object_Filler_v2.4
22	Net_10.10.21.0	net	10.10.21.0	255.255.255.0	blue	192.168.100.102	ngxr62	Hide	Created_by_Object_Filler_v2.4
23	Net_10.10.22.0	net	10.10.22.0	255.255.255.0	blue	192.168.100.103	ngxr62	Hide	Created_by_Object_Filler_v2.4
24	Net_10.10.23.0	net	10.10.23.0	255.255.255.0	blue	192.168.100.104	ngxr62	Hide	Created_by_Object_Filler_v2.4
25	Net_10.10.24.0	net	10.10.24.0	255.255.255.0	blue	192.168.100.105	ngxr62	Hide	Created_by_Object_Filler_v2.4
26	Net_10.10.25.0	net	10.10.25.0	255.255.255.0	blue	192.168.100.106	ngxr62	Hide	Created_by_Object_Filler_v2.4
27	Net_10.10.26.0	net	10.10.26.0	255.255.255.0	blue	192.168.100.107	ngxr62	Hide	Created_by_Object_Filler_v2.4
28	Net_10.10.27.0	net	10.10.27.0	255.255.255.0	blue	192.168.100.108	ngxr62	Hide	Created_by_Object_Filler_v2.4

Interesting is that you have evenly distributed the IP addresses of your NATting pool IP address, so you don't "load" everything on just one IP address (check that on the "F" column). Even though Check Point VPN-1 can Hide NAT around 4 billion IP addresses behind 1 NATting IP address, sometimes you may wish to distribute the "NATting load" (for any reason) among several addresses. Using Object Filler makes that easier…

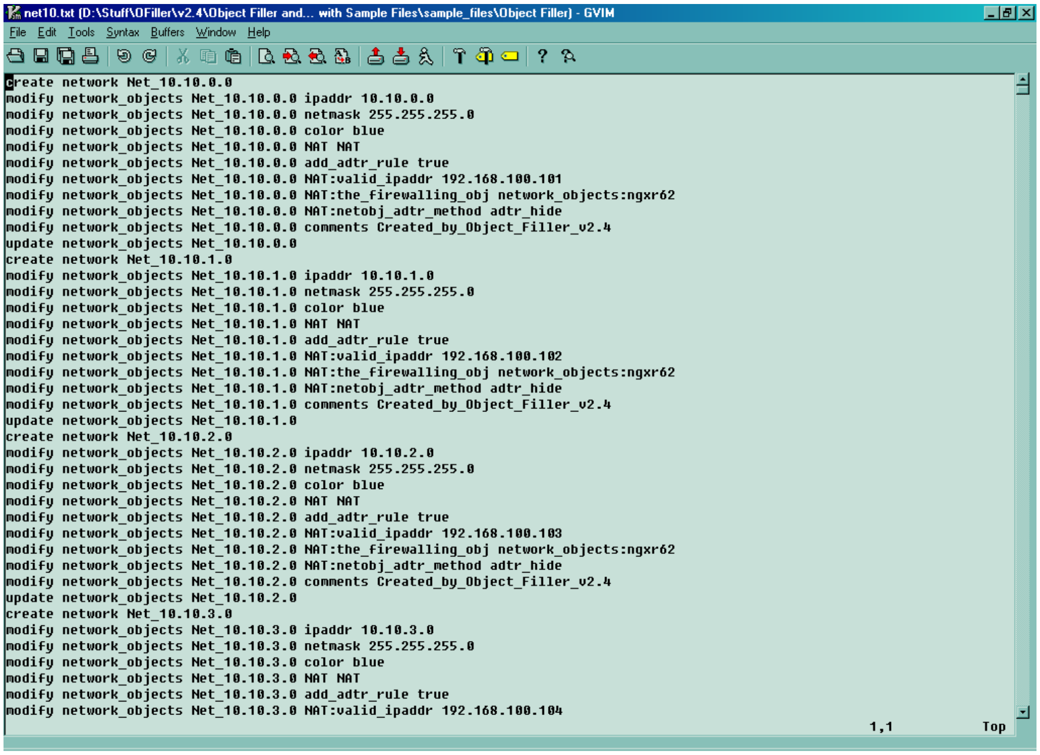
Ok. This looks fair enough. Now let's build the DBedit commands to we can actually stop playing games and begin doing the real stuff.

First, let's tell Object Filler to write real DBedit commands. Just substitute "-a" with "-o" and the filename as well. I'll change just the extension from *.csv* to *.txt*:

```
D:\Stuff\OFiller\V2.4>ofiller -s 10.10.0.0 -d 10.10.255.25 -m 24 -c blue -t net
-ns 192.168.100.101 -nd 192.168.100.110 -nm 24 -b ngxr62 -o net10.txt
Unofficial/Unsupported Object Filler V2.4  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
.....................................................
===============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 10 NATting IPs, 65306 possible objects and/or rules.
Found 256 total valid (or successfully processed) objects/rules.
Found 256 total NATted records.
-------------------------------------------------------------------------------
```

```
Total successfully processed Networks = 256
------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
==============================================================================
Task done successfully! - Thank you for using Object Filler V2.4!
```

Let's see what the output file looks like with a Text Editor


net10.txt (D:\Stuff\OFiller\v2.4\Object Filler and... with Sample Files\sample_files\Object Filler) - GVIM
File
Edit
Tools
Syntax
Buffers
Window
Help

```
create network Net_10.10.0.0
modify network_objects Net_10.10.0.0 ipaddr 10.10.0.0
modify network_objects Net_10.10.0.0 netmask 255.255.255.0
modify network_objects Net_10.10.0.0 color blue
modify network_objects Net_10.10.0.0 NAT NAT
modify network_objects Net_10.10.0.0 add_adtr_rule true
modify network_objects Net_10.10.0.0 NAT:valid_ipaddr 192.168.100.101
modify network_objects Net_10.10.0.0 NAT:the_firewalling_obj network_objects:ngxr62
modify network_objects Net_10.10.0.0 NAT:netobj_adtr_method adtr_hide
modify network_objects Net_10.10.0.0 comments Created_by_Object_Filler_v2.4
update network_objects Net_10.10.0.0
create network Net_10.10.1.0
modify network_objects Net_10.10.1.0 ipaddr 10.10.1.0
modify network_objects Net_10.10.1.0 netmask 255.255.255.0
modify network_objects Net_10.10.1.0 color blue
modify network_objects Net_10.10.1.0 NAT NAT
modify network_objects Net_10.10.1.0 add_adtr_rule true
modify network_objects Net_10.10.1.0 NAT:valid_ipaddr 192.168.100.102
modify network_objects Net_10.10.1.0 NAT:the_firewalling_obj network_objects:ngxr62
modify network_objects Net_10.10.1.0 NAT:netobj_adtr_method adtr_hide
modify network_objects Net_10.10.1.0 comments Created_by_Object_Filler_v2.4
update network_objects Net_10.10.1.0
create network Net_10.10.2.0
modify network_objects Net_10.10.2.0 ipaddr 10.10.2.0
modify network_objects Net_10.10.2.0 netmask 255.255.255.0
modify network_objects Net_10.10.2.0 color blue
modify network_objects Net_10.10.2.0 NAT NAT
modify network_objects Net_10.10.2.0 add_adtr_rule true
modify network_objects Net_10.10.2.0 NAT:valid_ipaddr 192.168.100.103
modify network_objects Net_10.10.2.0 NAT:the_firewalling_obj network_objects:ngxr62
modify network_objects Net_10.10.2.0 NAT:netobj_adtr_method adtr_hide
modify network_objects Net_10.10.2.0 comments Created_by_Object_Filler_v2.4
update network_objects Net_10.10.2.0
create network Net_10.10.3.0
modify network_objects Net_10.10.3.0 ipaddr 10.10.3.0
modify network_objects Net_10.10.3.0 netmask 255.255.255.0
modify network_objects Net_10.10.3.0 color blue
modify network_objects Net_10.10.3.0 NAT NAT
modify network_objects Net_10.10.3.0 add_adtr_rule true
modify network_objects Net_10.10.3.0 NAT:valid_ipaddr 192.168.100.104
                                                                1,1          Top
```

Couldn't be uglier… right? – all that is the DBedit script created by Object Filler to build all the objects.

Now, let's bring the file to the SecurePlatform machine. As a reminder, we have an FTP server in the Windows machine (in my case, that's 10.20.30.76) and will use the FTP client available with SecurePlatform. Please note that the transfer is done using ASCII mode. This is very important. Let's do it:

```
[Expert@ngxr62]# pwd
/home/admin
[Expert@ngxr62]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
```

```
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> get net10.txt
local: net10.txt remote: net10.txt
227 Entering passive mode (10,20,30,76,4,70)
125 Using existing data connection
################################################################################
################################################################################
#####
226 Closing data connection; File transfer successful.
168408 bytes received in 0.372 secs (4.4e+02 Kbytes/sec)
ftp> bye
221 Service closing control connection
[Expert@ngxr62]# ls -la net10.txt
-rw-rw----    1 root     root       165592 Nov 13 16:20 net10.txt
```

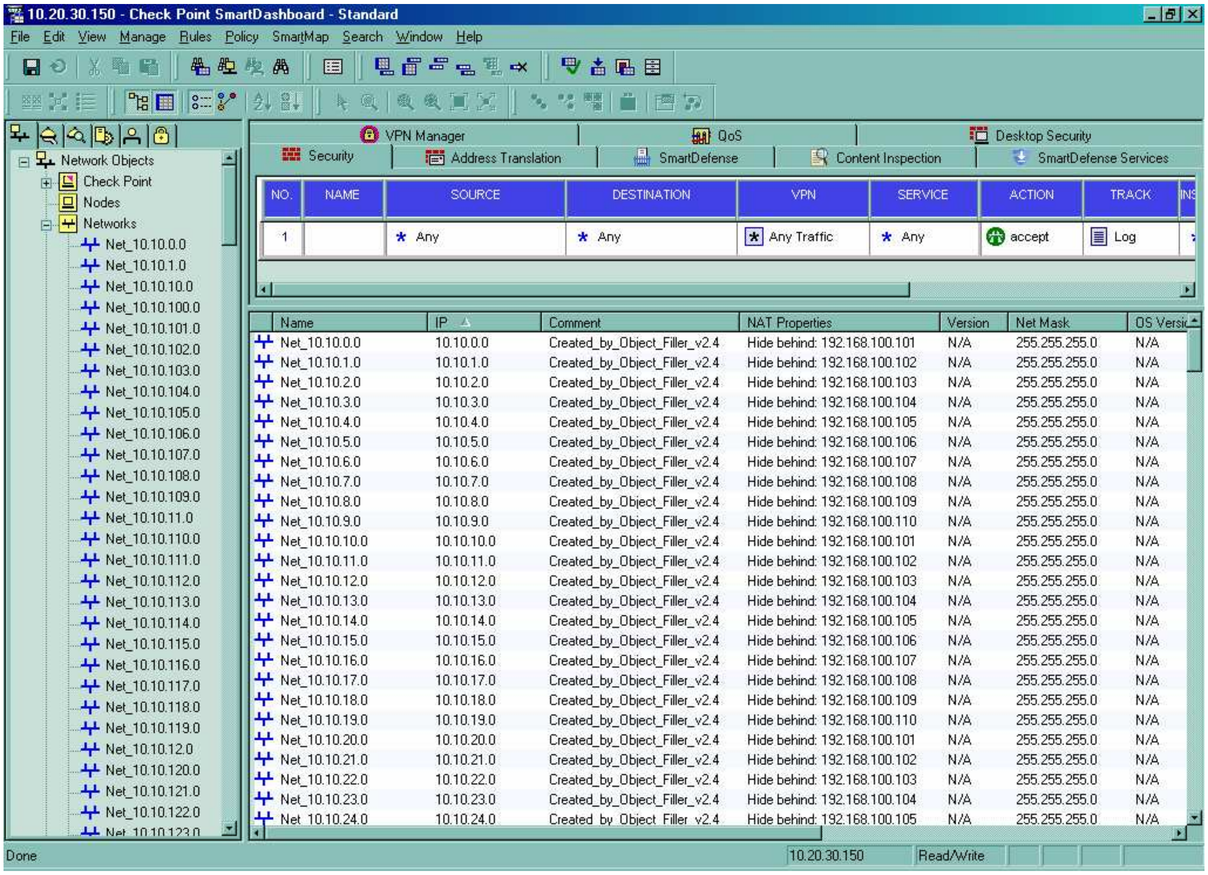
Okay, let’s import then the information to the SmartCenter. First make sure nobody is logged in locking the objects database. Remember you can use `cpstat mg` to check that out…

Then let’s tell DBedit to take the commands from the *net10.txt* file. Like this:

```
[Expert@ngxr62]# dbedit -f net10.txt -s localhost -u admin
Enter Administrator Password: secret
Net_10.10.0.0 updated successfully.
Net_10.10.1.0 updated successfully.
   .
   .
   .
Net_10.10.254.0 updated successfully.
Net_10.10.255.0 updated successfully.
[Expert@ngxr62]#
```

Notice that the difference this time, was that we used the “`-f net10.txt`” option, to tell DBedit to read the commands from a file. The time it takes to import the objects depends on your resources, but should take around one minute.

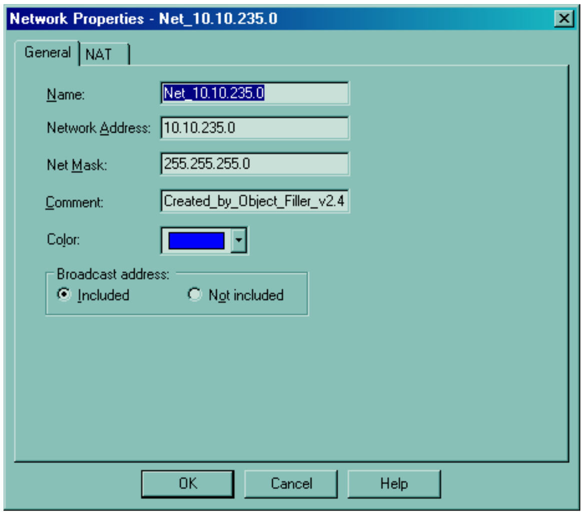
Let’s take a look in the SmartDashboard to see what the objects look like.



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INS
1		* Any	* Any	Any Traffic	* Any	accept	Log	

Name	IP	Comment	NAT Properties	Version	Net Mask	OS Versi
Net_10.10.0.0	10.10.0.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.101	N/A	255.255.255.0	N/A
Net_10.10.1.0	10.10.1.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.102	N/A	255.255.255.0	N/A
Net_10.10.2.0	10.10.2.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.103	N/A	255.255.255.0	N/A
Net_10.10.3.0	10.10.3.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.104	N/A	255.255.255.0	N/A
Net_10.10.4.0	10.10.4.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.105	N/A	255.255.255.0	N/A
Net_10.10.5.0	10.10.5.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.106	N/A	255.255.255.0	N/A
Net_10.10.6.0	10.10.6.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.107	N/A	255.255.255.0	N/A
Net_10.10.7.0	10.10.7.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.108	N/A	255.255.255.0	N/A
Net_10.10.8.0	10.10.8.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.109	N/A	255.255.255.0	N/A
Net_10.10.9.0	10.10.9.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.110	N/A	255.255.255.0	N/A
Net_10.10.10.0	10.10.10.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.101	N/A	255.255.255.0	N/A
Net_10.10.11.0	10.10.11.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.102	N/A	255.255.255.0	N/A
Net_10.10.12.0	10.10.12.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.103	N/A	255.255.255.0	N/A
Net_10.10.13.0	10.10.13.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.104	N/A	255.255.255.0	N/A
Net_10.10.14.0	10.10.14.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.105	N/A	255.255.255.0	N/A
Net_10.10.15.0	10.10.15.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.106	N/A	255.255.255.0	N/A
Net_10.10.16.0	10.10.16.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.107	N/A	255.255.255.0	N/A
Net_10.10.17.0	10.10.17.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.108	N/A	255.255.255.0	N/A
Net_10.10.18.0	10.10.18.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.109	N/A	255.255.255.0	N/A
Net_10.10.19.0	10.10.19.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.110	N/A	255.255.255.0	N/A
Net_10.10.20.0	10.10.20.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.101	N/A	255.255.255.0	N/A
Net_10.10.21.0	10.10.21.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.102	N/A	255.255.255.0	N/A
Net_10.10.22.0	10.10.22.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.103	N/A	255.255.255.0	N/A
Net_10.10.23.0	10.10.23.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.104	N/A	255.255.255.0	N/A
Net_10.10.24.0	10.10.24.0	Created_by_Object_Filler_v2.4	Hide behind: 192.168.100.105	N/A	255.255.255.0	N/A

10.20.30.150 Read/Write



Network Properties - Net_10.10.235.0

Name: Net_10.10.235.0

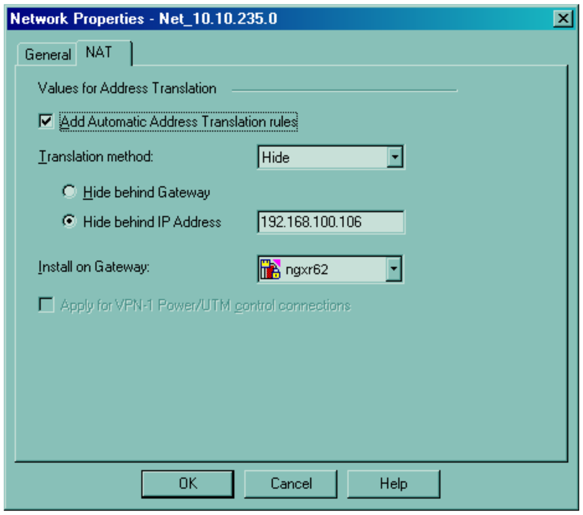
Network Address: 10.10.235.0

Net Mask: 255.255.255.0

Comment: Created_by_Object_Filler_v2.4

Color:

Broadcast address: ● Included ○ Not included



Network Properties - Net_10.10.235.0

Values for Address Translation

☑ Add Automatic Address Translation rules

Translation method: Hide

○ Hide behind Gateway

● Hide behind IP Address 192.168.100.106

Install on Gateway: ngxr62

□ Apply for VPN-1 Power/UTM control connections

Notice on the list the IP Addresses used to NAT the network objects just created.

Nice job all around, huh?

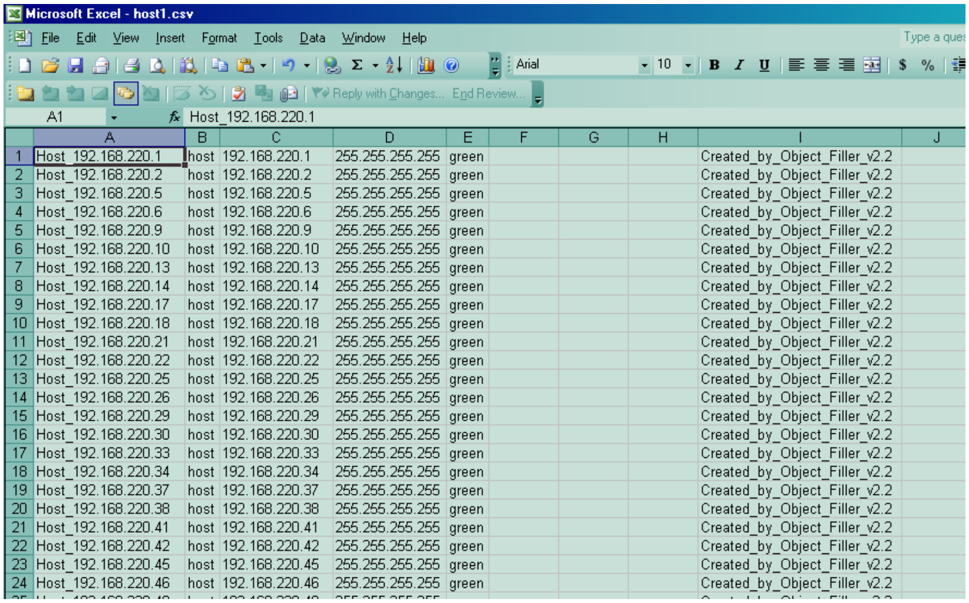
5.1.2 Creating consecutive hosts

Creating consecutive hosts is just as easy as building consecutive networks. Just as Object Filler could tell when a net should be processed, Object Filler can tell as well when a host inside a net should be processed, using the netmask bits provided. Let’s see how this works.

Let’s assume you have the 192.168.220.0/24 network and want to subnet it with a 30 bits netmask. This means small subnets with 2 hosts each one. Let’s ask Object Filler to process just the hosts for each subnet. Since you already know the syntax, this time the syntax won’t be explained fully:

```
D:\Stuff\OFiller\V2.2>ofiller -t hosts -s 192.168.220.0 -d 192.168.220.255 -m 30
 -c green -a host1.csv
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
==============================================================================
.........................
==============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 256 possible objects and/or rules.
Found 128 total valid (or successfully processed) objects/rules.
------------------------------------------------------------------------------
Total successfully processed Hosts = 128
------------------------------------------------------------------------------
Please review that all CSV output information was written correctly.
Please remember CSV information is for reviewing only. For importing into
SmartCenter you need to use DBedit mode (-o switch).
==============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

Let’s take a look at the output with Microsoft Excel.


Microsoft Excel - host1.csv
File Edit View Insert Format Tools Data Window Help
Type a ques
Arial
10
Reply with Changes... End Review...
A1
Host_192.168.220.1

	A	B	C	D	E	F	G	H	I	J
1	Host_192.168.220.1	host	192.168.220.1	255.255.255.255	green				Created_by_Object_Filler_v2.2	
2	Host_192.168.220.2	host	192.168.220.2	255.255.255.255	green				Created_by_Object_Filler_v2.2	
3	Host_192.168.220.5	host	192.168.220.5	255.255.255.255	green				Created_by_Object_Filler_v2.2	
4	Host_192.168.220.6	host	192.168.220.6	255.255.255.255	green				Created_by_Object_Filler_v2.2	
5	Host_192.168.220.9	host	192.168.220.9	255.255.255.255	green				Created_by_Object_Filler_v2.2	
6	Host_192.168.220.10	host	192.168.220.10	255.255.255.255	green				Created_by_Object_Filler_v2.2	
7	Host_192.168.220.13	host	192.168.220.13	255.255.255.255	green				Created_by_Object_Filler_v2.2	
8	Host_192.168.220.14	host	192.168.220.14	255.255.255.255	green				Created_by_Object_Filler_v2.2	
9	Host_192.168.220.17	host	192.168.220.17	255.255.255.255	green				Created_by_Object_Filler_v2.2	
10	Host_192.168.220.18	host	192.168.220.18	255.255.255.255	green				Created_by_Object_Filler_v2.2	
11	Host_192.168.220.21	host	192.168.220.21	255.255.255.255	green				Created_by_Object_Filler_v2.2	
12	Host_192.168.220.22	host	192.168.220.22	255.255.255.255	green				Created_by_Object_Filler_v2.2	
13	Host_192.168.220.25	host	192.168.220.25	255.255.255.255	green				Created_by_Object_Filler_v2.2	
14	Host_192.168.220.26	host	192.168.220.26	255.255.255.255	green				Created_by_Object_Filler_v2.2	
15	Host_192.168.220.29	host	192.168.220.29	255.255.255.255	green				Created_by_Object_Filler_v2.2	
16	Host_192.168.220.30	host	192.168.220.30	255.255.255.255	green				Created_by_Object_Filler_v2.2	
17	Host_192.168.220.33	host	192.168.220.33	255.255.255.255	green				Created_by_Object_Filler_v2.2	
18	Host_192.168.220.34	host	192.168.220.34	255.255.255.255	green				Created_by_Object_Filler_v2.2	
19	Host_192.168.220.37	host	192.168.220.37	255.255.255.255	green				Created_by_Object_Filler_v2.2	
20	Host_192.168.220.38	host	192.168.220.38	255.255.255.255	green				Created_by_Object_Filler_v2.2	
21	Host_192.168.220.41	host	192.168.220.41	255.255.255.255	green				Created_by_Object_Filler_v2.2	
22	Host_192.168.220.42	host	192.168.220.42	255.255.255.255	green				Created_by_Object_Filler_v2.2	
23	Host_192.168.220.45	host	192.168.220.45	255.255.255.255	green				Created_by_Object_Filler_v2.2	
24	Host_192.168.220.46	host	192.168.220.46	255.255.255.255	green				Created_by_Object_Filler_v2.2	

Please notice that NOT all the hosts were built, but just the ones that made sense according to the bit mask we provided, which was 30 bits. For example, 192.168.220.3 is a broadcast address and 192.168.220.4 is a network address when subnetting to 30 bits, so those objects were not created. However, the netmask for the object itself is 32 bits, as we're talking about a host…

Okay, seems fair enough. Now let's build the DBedit commands.

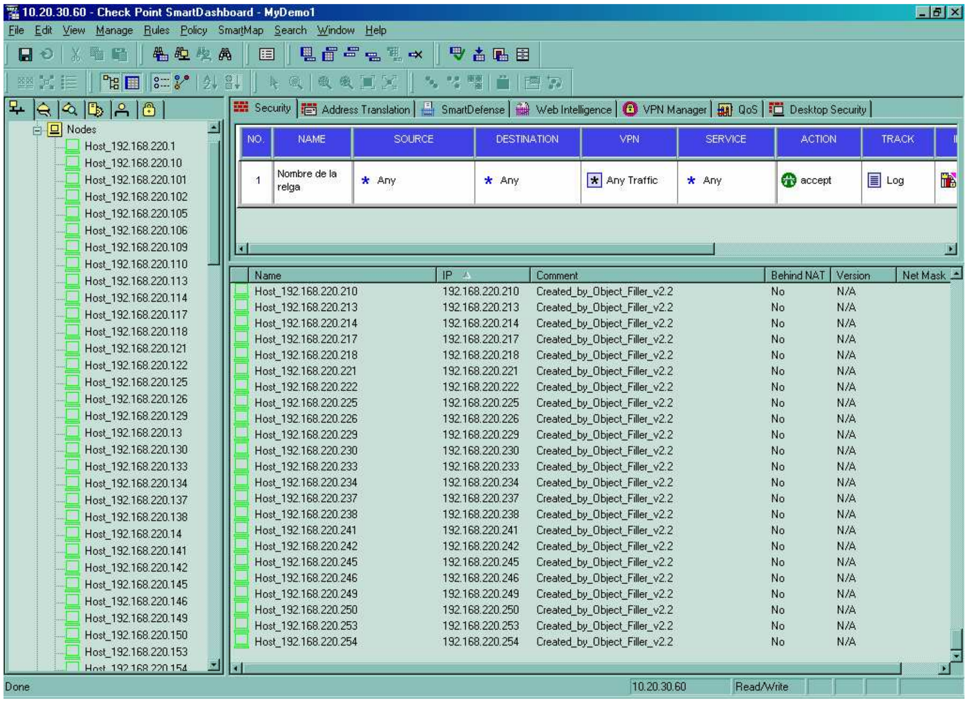
```
D:\Stuff\OFiller\V2.2>ofiller -t hosts -s 192.168.220.0 -d 192.168.220.255 -m 30
 -c green -o host1.dbedit
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
..........................
===============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 256 possible objects and/or rules.
Found 128 total valid (or successfully processed) objects/rules.
-------------------------------------------------------------------------------
Total successfully processed Hosts = 128
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

Then let's import these objects into SmartCenter. Again, don't forget to check that there is no lock over the objects database using `cpstat mg`:

```
[Expert@ngxr62]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> get host1.dbedit
local: host1.dbedit remote: host1.dbedit
227 Entering passive mode (10,20,30,76,4,78)
125 Using existing data connection
#####################################
226 Closing data connection; File transfer successful.
37302 bytes received in 0.0806 secs (4.5e+02 Kbytes/sec)
ftp> bye
221 Service closing control connection
[Expert@ngxr62]# ls -la host1.dbedit
-rw-rw----    1 root     root        36662 Nov 13 17:00 host1.dbedit
[Expert@ngxr62]# dbedit -s localhost -u admin -f host1.dbedit
Enter Administrator Password: secret
Host_192.168.220.1 updated successfully.
Host_192.168.220.2 updated successfully.
```

```
      .
      .
      .
Host_192.168.220.253 updated successfully.
Host_192.168.220.254 updated successfully.
```

Let’s take a look now in the SmartDashboard.



10.20.30.60 - Check Point SmartDashboard - MyDemo1

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK
1	Nombre de la relga	* Any	* Any	Any Traffic	* Any	accept	Log

Nodes
Host_192.168.220.1
Host_192.168.220.10
Host_192.168.220.101
Host_192.168.220.102
Host_192.168.220.105
Host_192.168.220.106
Host_192.168.220.109
Host_192.168.220.110
Host_192.168.220.113
Host_192.168.220.114
Host_192.168.220.117
Host_192.168.220.118
Host_192.168.220.121
Host_192.168.220.122
Host_192.168.220.125
Host_192.168.220.126
Host_192.168.220.129
Host_192.168.220.13
Host_192.168.220.130
Host_192.168.220.133
Host_192.168.220.134
Host_192.168.220.137
Host_192.168.220.138
Host_192.168.220.14
Host_192.168.220.141
Host_192.168.220.142
Host_192.168.220.145
Host_192.168.220.146
Host_192.168.220.149
Host_192.168.220.150
Host_192.168.220.153
Host_192.168.220.154

Name	IP	Comment	Behind NAT	Version	Net Mask
Host_192.168.220.210	192.168.220.210	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.213	192.168.220.213	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.214	192.168.220.214	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.217	192.168.220.217	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.218	192.168.220.218	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.221	192.168.220.221	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.222	192.168.220.222	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.225	192.168.220.225	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.226	192.168.220.226	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.229	192.168.220.229	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.230	192.168.220.230	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.233	192.168.220.233	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.234	192.168.220.234	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.237	192.168.220.237	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.238	192.168.220.238	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.241	192.168.220.241	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.242	192.168.220.242	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.245	192.168.220.245	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.246	192.168.220.246	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.249	192.168.220.249	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.250	192.168.220.250	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.253	192.168.220.253	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.254	192.168.220.254	Created_by_Object_Filler_v2.2	No	N/A	

Done
10.20.30.60
Read/Write

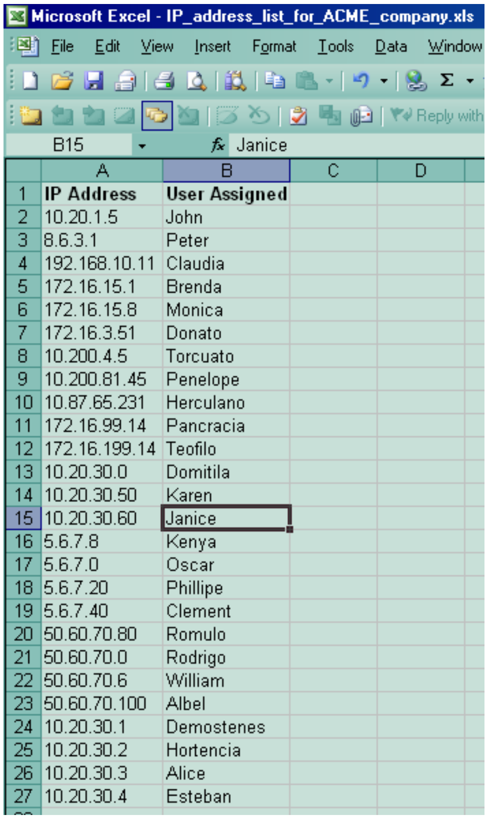
All seems to be fine. So, we’re done! – Did you take your time? ;-)

5.2 Creating a list file to import objects

Sometimes there’s already a list of IPs that belong to a group of people, since this is a quite common *IP management* system ;-)

Perhaps such a list doesn’t exist yet. However, your customers (or you) are willing to create it before populating the SmartCenter, as it eases the process of creating rules later – since it gives a clearer view of what you’re configuring.

Let’s view how the list may look…

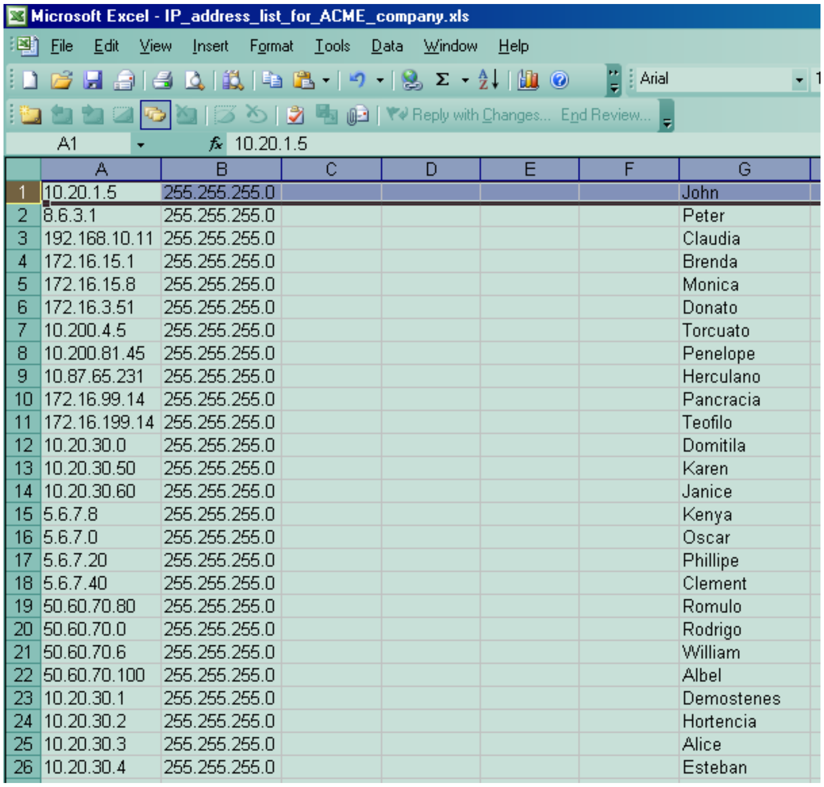


B15 fx Janice

	A	B	C	D
1	**IP Address**	**User Assigned**		
2	10.20.1.5	John		
3	8.6.3.1	Peter		
4	192.168.10.11	Claudia		
5	172.16.15.1	Brenda		
6	172.16.15.8	Monica		
7	172.16.3.51	Donato		
8	10.200.4.5	Torcuato		
9	10.200.81.45	Penelope		
10	10.87.65.231	Herculano		
11	172.16.99.14	Pancracia		
12	172.16.199.14	Teofilo		
13	10.20.30.0	Domitila		
14	10.20.30.50	Karen		
15	10.20.30.60	Janice		
16	5.6.7.8	Kenya		
17	5.6.7.0	Oscar		
18	5.6.7.20	Phillipe		
19	5.6.7.40	Clement		
20	50.60.70.80	Romulo		
21	50.60.70.0	Rodrigo		
22	50.60.70.6	William		
23	50.60.70.100	Albel		
24	10.20.30.1	Demostenes		
25	10.20.30.2	Hortencia		
26	10.20.30.3	Alice		
27	10.20.30.4	Esteban		

Let’s suppose that’s just the sample of the first department and you have 10 more departments whose *IP management system* is the same: A Microsoft Excel sheet. And now, you are tasked on build a host object for each one of the IPs that are assigned... What a boring duty!

Well, Object Filler can help again. First, let’s modify slightly the spreadsheet to make it look like this:


Microsoft Excel - IP_address_list_for_ACME_company.xls
File Edit View Insert Format Tools Data Window Help
Arial
Reply with Changes... End Review...

A1 10.20.1.5

	A	B	C	D	E	F	G
1	10.20.1.5	255.255.255.0					John
2	8.6.3.1	255.255.255.0					Peter
3	192.168.10.11	255.255.255.0					Claudia
4	172.16.15.1	255.255.255.0					Brenda
5	172.16.15.8	255.255.255.0					Monica
6	172.16.3.51	255.255.255.0					Donato
7	10.200.4.5	255.255.255.0					Torcuato
8	10.200.81.45	255.255.255.0					Penelope
9	10.87.65.231	255.255.255.0					Herculano
10	172.16.99.14	255.255.255.0					Pancracia
11	172.16.199.14	255.255.255.0					Teofilo
12	10.20.30.0	255.255.255.0					Domitila
13	10.20.30.50	255.255.255.0					Karen
14	10.20.30.60	255.255.255.0					Janice
15	5.6.7.8	255.255.255.0					Kenya
16	5.6.7.0	255.255.255.0					Oscar
17	5.6.7.20	255.255.255.0					Phillipe
18	5.6.7.40	255.255.255.0					Clement
19	50.60.70.80	255.255.255.0					Romulo
20	50.60.70.0	255.255.255.0					Rodrigo
21	50.60.70.6	255.255.255.0					William
22	50.60.70.100	255.255.255.0					Albel
23	10.20.30.1	255.255.255.0					Demostenes
24	10.20.30.2	255.255.255.0					Hortencia
25	10.20.30.3	255.255.255.0					Alice
26	10.20.30.4	255.255.255.0					Esteban

It took me less than 30 seconds inserting columns and copy-pasting the netmask value in all the cells! ;-)

Please note than on rows 12, 16 and 20 there are IP addresses that under this netmask would be networks and not hosts. I want to point this out, so in case they are really hosts and not networks, then you should put the correct netmask. For our example, we'll assume they really are subnets and not hosts.

Once you have the file on this format, save it as a CSV file. With Microsoft Excel 2003, this is accomplished doing this: Go to *File Menu*, then *Save As*, then under *Save as type* choose *CSV (Comma delimited) (*.csv)*. If you use some other spreadsheet program, you probably have the option of saving the file as CSV under the *Save As* option as well. A couple of warning messages will be shown about losing the native format, just accept them as it is fine. The file is now there

```
D:\Stuff\OFiller\V2.4>dir IP_address_list_for_ACME_company.csv
 Volume in drive D is Data
 Volume Serial Number is 88AE-1253

 Directory of D:\Stuff\OFiller\V2.2

13/11/2005  05:49 p.m.               987 IP_address_list_for_ACME_company.csv
               1 File(s)            987 bytes
               0 Dir(s)     751,386,624 bytes free
```

Now, use Object Filler over the file you just created

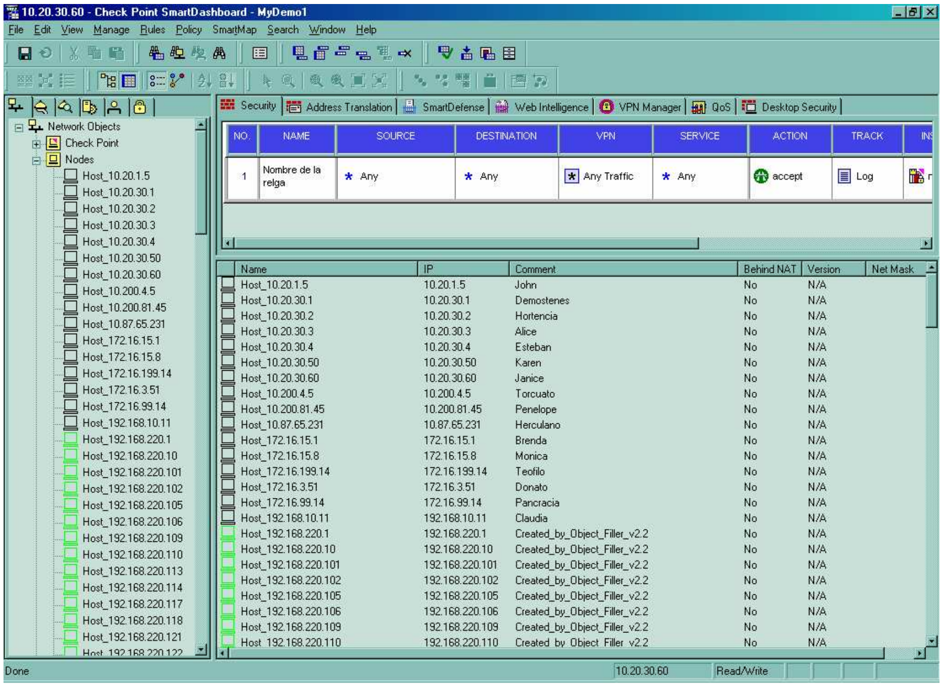
```
D:\Stuff\OFiller\V2.2>ofiller -f IP_address_list_for_ACME_company.csv -i list -o
 mylist.dbedit
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
.....
===============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 52 possible objects and/or rules.
Found 26 total valid (or successfully processed) objects/rules.
-------------------------------------------------------------------------------
Total successfully processed Hosts = 23
Total successfully processed Networks = 3
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

And then transfer the file to the SmartCenter, and import it using DBedit. You already know how to do this.

```
[Expert@ngxr62]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> get mylist.dbedit
local: mylist.dbedit remote: mylist.dbedit
227 Entering passive mode (10,20,30,76,4,86)
125 Using existing data connection
#######
226 Closing data connection; File transfer successful.
6529 bytes received in 0.0306 secs (2.1e+02 Kbytes/sec)
ftp> bye
221 Service closing control connection
[Expert@ngxr62]# ls -la mylist.dbedit
-rw-rw----    1 root     root         6396 Nov 13 18:54 mylist.dbedit
[Expert@ngxr62]# dbedit -f mylist.dbedit -s localhost -u admin
Enter Administrator Password: secret
Host_10.20.1.5 updated successfully.
Host_8.6.3.1 updated successfully.
    .
    .
    .
Host_10.20.30.3 updated successfully.
```

```
Host_10.20.30.4 updated successfully.
[Expert@ngxr62]#
```

Of course, after this, it goes the evidential screenshot. Where the heck are my new objects? Here!



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INS
1	Nombre de la relga	* Any	* Any	Any Traffic	* Any	accept	Log	r

Name	IP	Comment	Behind NAT	Version	Net Mask
Host_10.20.1.5	10.20.1.5	John	No	N/A	
Host_10.20.30.1	10.20.30.1	Demostenes	No	N/A	
Host_10.20.30.2	10.20.30.2	Hortencia	No	N/A	
Host_10.20.30.3	10.20.30.3	Alice	No	N/A	
Host_10.20.30.4	10.20.30.4	Esteban	No	N/A	
Host_10.20.30.50	10.20.30.50	Karen	No	N/A	
Host_10.20.30.60	10.20.30.60	Janice	No	N/A	
Host_10.200.4.5	10.200.4.5	Torcuato	No	N/A	
Host_10.200.81.45	10.200.81.45	Penelope	No	N/A	
Host_10.87.65.231	10.87.65.231	Herculano	No	N/A	
Host_172.16.15.1	172.16.15.1	Brenda	No	N/A	
Host_172.16.15.8	172.16.15.8	Monica	No	N/A	
Host_172.16.199.14	172.16.199.14	Teofilo	No	N/A	
Host_172.16.3.51	172.16.3.51	Donato	No	N/A	
Host_172.16.99.14	172.16.99.14	Pancracia	No	N/A	
Host_192.168.10.11	192.168.10.11	Claudia	No	N/A	
Host_192.168.220.1	192.168.220.1	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.10	192.168.220.10	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.101	192.168.220.101	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.102	192.168.220.102	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.105	192.168.220.105	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.106	192.168.220.106	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.109	192.168.220.109	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.110	192.168.220.110	Created_by_Object_Filler_v2.2	No	N/A	

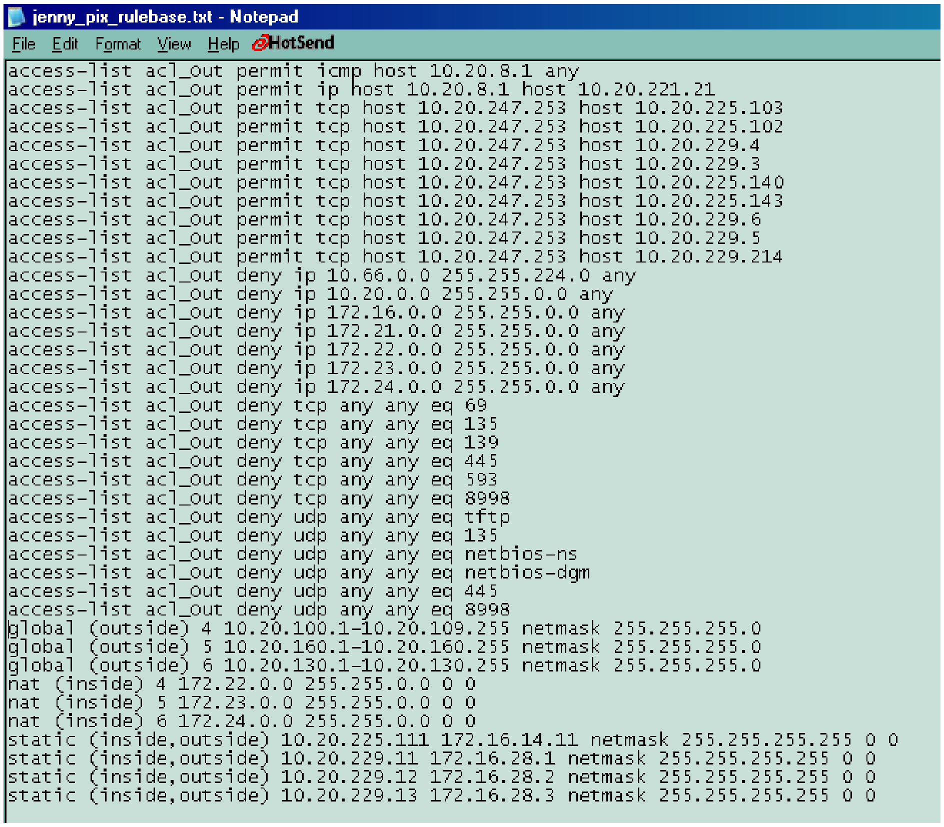
By the way, the color of the newer object was left as default, which is black. This way you can identify them…

5.3 Importing objects from Cisco PIX/ASA/FWSM and Cisco Routers configurations

Okay. Now the heavy stuff. This is for really initiated people. For Wizards. For masters and gurus. This is… for people reading this document…; -)

Let's say you got a customer that has a Cisco PIX/ASA/FWSM firewall and she's going to change to Check Point. The problem is, the rulebase is quite big. Lots of objects, lots of rules… and seems is not going to be easy…

Let's take a look (at a sample) of what she might have:


jenny_pix_rulebase.txt - Notepad
File Edit Format View Help HotSend

```
access-list acl_out permit icmp host 10.20.8.1 any
access-list acl_out permit ip host 10.20.8.1 host 10.20.221.21
access-list acl_out permit tcp host 10.20.247.253 host 10.20.225.103
access-list acl_out permit tcp host 10.20.247.253 host 10.20.225.102
access-list acl_out permit tcp host 10.20.247.253 host 10.20.229.4
access-list acl_out permit tcp host 10.20.247.253 host 10.20.229.3
access-list acl_out permit tcp host 10.20.247.253 host 10.20.225.140
access-list acl_out permit tcp host 10.20.247.253 host 10.20.225.143
access-list acl_out permit tcp host 10.20.247.253 host 10.20.229.6
access-list acl_out permit tcp host 10.20.247.253 host 10.20.229.5
access-list acl_out permit tcp host 10.20.247.253 host 10.20.229.214
access-list acl_out deny ip 10.66.0.0 255.255.224.0 any
access-list acl_out deny ip 10.20.0.0 255.255.0.0 any
access-list acl_out deny ip 172.16.0.0 255.255.0.0 any
access-list acl_out deny ip 172.21.0.0 255.255.0.0 any
access-list acl_out deny ip 172.22.0.0 255.255.0.0 any
access-list acl_out deny ip 172.23.0.0 255.255.0.0 any
access-list acl_out deny ip 172.24.0.0 255.255.0.0 any
access-list acl_out deny tcp any any eq 69
access-list acl_out deny tcp any any eq 135
access-list acl_out deny tcp any any eq 139
access-list acl_out deny tcp any any eq 445
access-list acl_out deny tcp any any eq 593
access-list acl_out deny tcp any any eq 8998
access-list acl_out deny udp any any eq tftp
access-list acl_out deny udp any any eq 135
access-list acl_out deny udp any any eq netbios-ns
access-list acl_out deny udp any any eq netbios-dgm
access-list acl_out deny udp any any eq 445
access-list acl_out deny udp any any eq 8998
global (outside) 4 10.20.100.1-10.20.109.255 netmask 255.255.255.0
global (outside) 5 10.20.160.1-10.20.160.255 netmask 255.255.255.0
global (outside) 6 10.20.130.1-10.20.130.255 netmask 255.255.255.0
nat (inside) 4 172.22.0.0 255.255.0.0 0 0
nat (inside) 5 172.23.0.0 255.255.0.0 0 0
nat (inside) 6 172.24.0.0 255.255.0.0 0 0
static (inside,outside) 10.20.225.111 172.16.14.11 netmask 255.255.255.255 0 0
static (inside,outside) 10.20.229.11 172.16.28.1 netmask 255.255.255.255 0 0
static (inside,outside) 10.20.229.12 172.16.28.2 netmask 255.255.255.255 0 0
static (inside,outside) 10.20.229.13 172.16.28.3 netmask 255.255.255.255 0 0
```

And you complain and beg: “Jenny, that’s terribly awful!, please don’t make me go through this step”. While she replies: “I’m sorry, you behaved bad and it is your punishment to convert all that into a more cute format”. And finally, you mumble: “All right! I’ll do it!”

Don’t worry desperate firewall manager! Object Filler is here to save you once again!

First let’s see that the file is there available for processing:

```
D:\Stuff\OFiller\V2.4>dir jenny_pix_rulebase.txt
 Volume in drive D is Data
 Volume Serial Number is 88AE-1253

 Directory of D:\Stuff\OFiller\V2.4

13/11/2005  06:15 p.m.             2,364 jenny_pix_rulebase.txt
               1 File(s)          2,364 bytes
               0 Dir(s)     748,654,592 bytes free
```

Alright. Now all you have to do is tell Object Filler you will be importing from a Cisco PIX configuration file and will do most of the work for you.

```
D:\Stuff\OFiller\V2.4>ofiller -f jenny_pix_rulebase.txt -i pix -a mypixsample1.c
sv
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
```

```
(c) 2003-2006 by Check Point Software Technologies, Inc.
================================================================================

Processing NATted Objects...
..
Processing Regular Objects...
...........

--------------------------------------------------------------------------------
Verifying existing records - creating unknown services
--------------------------------------------------------------------------------

--------------------------------------------------------------------------------
Total invalid objects processed = 19
--------------------------------------------------------------------------------
NOTE: These invalid objects are the ones that were not processed  successfully.
      Unless you used the No Objects or Policy Verification switch (-nopv);  on
      which case, the objects were processed  normally  as  if  they  were  not
      conflicting with something else. But there were still marked as invalid.
      Usually  this is a matter of duplicates: same IP address  with  different
      name, or same object name with different IP address.
      However,  the ones they were conflicting with (marked as  valid  records)
      were  processed with no further problems. If this was a processing  error
      of  any  kind,  you  can  always create  the  object  again  whithin  the
      Check Point SmartDashboard.
      Please run Object Filler again over the same source and use this time the
      "-v" (verbose) parameter. The  output  will  tell  you  in  the  object's
      processing line more details on why such objects failed.

================================================================================
*** Successfully imported configuration from Cisco PIX!
--------------------------------------------------------------------------------
It took 3.0 seconds of total processing time on QUIET Mode.
Processed 362 possible objects and/or rules.
Found 39 total valid (or successfully processed) objects/rules.
Found 7 total NATted records.
--------------------------------------------------------------------------------
Total successfully processed Hosts = 20
Total successfully processed Networks = 7
Total successfully processed TCP Service objects = 6
Total successfully processed UDP Service objects = 6
--------------------------------------------------------------------------------
Please review that all CSV output information was written correctly.
Please remember CSV information is for reviewing only. For importing into
SmartCenter you need to use DBedit mode (-o switch).
================================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

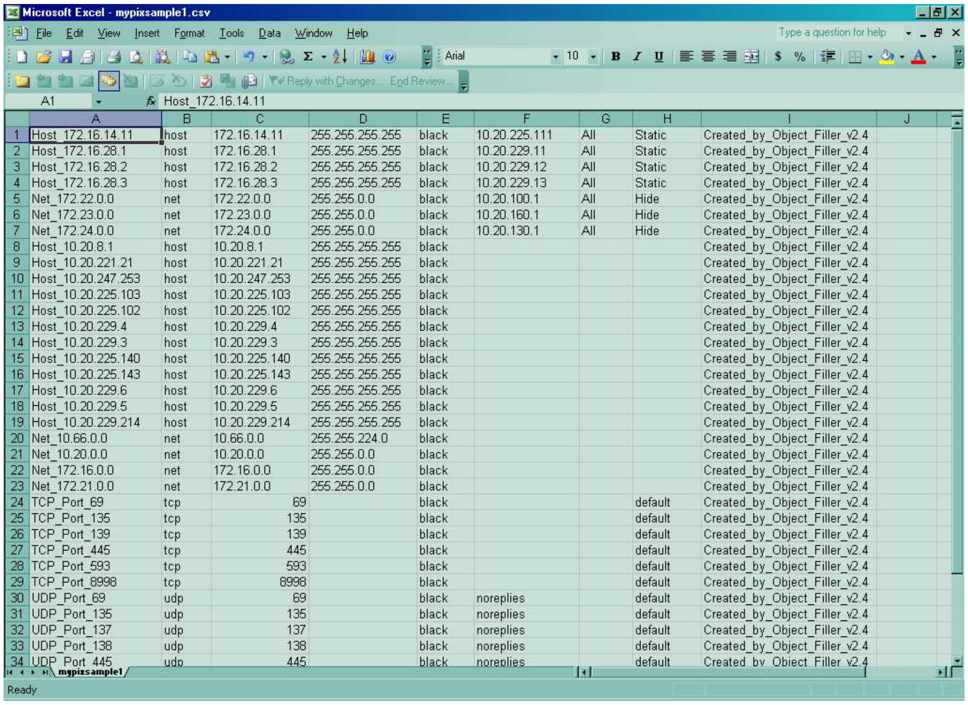
Some of this stuff is new and it deserves some explanation. Let’s see..

First, Object Filler parses twice the Cisco PIX configuration file looking for IP addresses and netmasks that make sense, so it can build objects. The first time looks for NAT statements, so it can properly do NAT of objects. Object Filler supports both Static and Hide NAT when importing Cisco PIX configurations. Then it parses the configuration again looking for other objects. Finally it builds services that are seen in the configuration.

Object Filler always avoids duplicates. So, if it sees an IP address in 2 different places, the first time builds an object for it and the second time reports a *duplicate*, or *invalid objects*. This is what the *invalid objects* report is about. If you want to know what duplicates were found, and in general more information on how the processing is done, is advised to use the verbose mode (-v *switch*) so you can see directly to your screen the inner details of the processing. The output could be really overwhelming with a big configuration file, but it pays off if you want to save yourself some pain later. Besides, there's nothing big enough that can't be enjoyable with a cup of good coffee or a glass of some good beer, right?

The last thing is that Object Filler has some internal mechanisms that try to detect if the configuration you are importing is really of the type you indicated. In this case, it tries to figure out if this is really a PIX configuration file. If it is not, it gives you a warning. If it is, it tells you the configuration was successfully imported, which was the case this time. Even though this mechanism is not 100% accurate with modified files, it is a very helpful aid and with regular configurations (this is, the ones that are not trimmed or tricked some how) should be accurate all the times.

Let's take a look on the resulting CSV File:



A1 Host_172.16.14.11

	A	B	C	D	E	F	G	H	I	J
1	Host_172.16.14.11	host	172.16.14.11	255.255.255.255	black	10.20.225.111	All	Static	Created_by_Object_Filler_v2.4	
2	Host_172.16.28.1	host	172.16.28.1	255.255.255.255	black	10.20.229.11	All	Static	Created_by_Object_Filler_v2.4	
3	Host_172.16.28.2	host	172.16.28.2	255.255.255.255	black	10.20.229.12	All	Static	Created_by_Object_Filler_v2.4	
4	Host_172.16.28.3	host	172.16.28.3	255.255.255.255	black	10.20.229.13	All	Static	Created_by_Object_Filler_v2.4	
5	Net_172.22.0.0	net	172.22.0.0	255.255.0.0	black	10.20.100.1	All	Hide	Created_by_Object_Filler_v2.4	
6	Net_172.23.0.0	net	172.23.0.0	255.255.0.0	black	10.20.160.1	All	Hide	Created_by_Object_Filler_v2.4	
7	Net_172.24.0.0	net	172.24.0.0	255.255.0.0	black	10.20.130.1	All	Hide	Created_by_Object_Filler_v2.4	
8	Host_10.20.8.1	host	10.20.8.1	255.255.255.255	black				Created_by_Object_Filler_v2.4	
9	Host_10.20.221.21	host	10.20.221.21	255.255.255.255	black				Created_by_Object_Filler_v2.4	
10	Host_10.20.247.253	host	10.20.247.253	255.255.255.255	black				Created_by_Object_Filler_v2.4	
11	Host_10.20.225.103	host	10.20.225.103	255.255.255.255	black				Created_by_Object_Filler_v2.4	
12	Host_10.20.225.102	host	10.20.225.102	255.255.255.255	black				Created_by_Object_Filler_v2.4	
13	Host_10.20.229.4	host	10.20.229.4	255.255.255.255	black				Created_by_Object_Filler_v2.4	
14	Host_10.20.229.3	host	10.20.229.3	255.255.255.255	black				Created_by_Object_Filler_v2.4	
15	Host_10.20.225.140	host	10.20.225.140	255.255.255.255	black				Created_by_Object_Filler_v2.4	
16	Host_10.20.225.143	host	10.20.225.143	255.255.255.255	black				Created_by_Object_Filler_v2.4	
17	Host_10.20.229.6	host	10.20.229.6	255.255.255.255	black				Created_by_Object_Filler_v2.4	
18	Host_10.20.229.5	host	10.20.229.5	255.255.255.255	black				Created_by_Object_Filler_v2.4	
19	Host_10.20.229.214	host	10.20.229.214	255.255.255.255	black				Created_by_Object_Filler_v2.4	
20	Net_10.66.0.0	net	10.66.0.0	255.255.224.0	black				Created_by_Object_Filler_v2.4	
21	Net_10.20.0.0	net	10.20.0.0	255.255.0.0	black				Created_by_Object_Filler_v2.4	
22	Net_172.16.0.0	net	172.16.0.0	255.255.0.0	black				Created_by_Object_Filler_v2.4	
23	Net_172.21.0.0	net	172.21.0.0	255.255.0.0	black				Created_by_Object_Filler_v2.4	
24	TCP_Port_69	tcp	69		black			default	Created_by_Object_Filler_v2.4	
25	TCP_Port_135	tcp	135		black			default	Created_by_Object_Filler_v2.4	
26	TCP_Port_139	tcp	139		black			default	Created_by_Object_Filler_v2.4	
27	TCP_Port_445	tcp	445		black			default	Created_by_Object_Filler_v2.4	
28	TCP_Port_593	tcp	593		black			default	Created_by_Object_Filler_v2.4	
29	TCP_Port_8998	tcp	8998		black			default	Created_by_Object_Filler_v2.4	
30	UDP_Port_69	udp	69		black	noreplies		default	Created_by_Object_Filler_v2.4	
31	UDP_Port_135	udp	135		black	noreplies		default	Created_by_Object_Filler_v2.4	
32	UDP_Port_137	udp	137		black	noreplies		default	Created_by_Object_Filler_v2.4	
33	UDP_Port_138	udp	138		black	noreplies		default	Created_by_Object_Filler_v2.4	
34	UDP_Port_445	udp	445		black	noreplies		default	Created_by_Object_Filler_v2.4	

As you can see, all the details are shown there. In a more cute and more readable format, as Jenny requested! ;-) As explained above, if you want to know even more details on the conversion, it is suggested to use the verbose mode (-v) of Object Filler to get such information.

Let's now finish the job producing the DBedit script.

```
D:\Stuff\OFiller\V2.4>ofiller -f jenny_pix_rulebase.txt -i pix -o mypix1.dbedit
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

Processing NATted Objects...
..
Processing Regular Objects...
...........

-------------------------------------------------------------------------------
Verifying existing records - creating unknown services
-------------------------------------------------------------------------------

-------------------------------------------------------------------------------
Total invalid objects processed = 19
-------------------------------------------------------------------------------
NOTE: These invalid objects are the ones that were not processed  successfully.
      Unless you used the No Objects or Policy Verification switch (-nopv);  on
      which case, the objects were processed  normally  as  if  they  were  not
      conflicting with something else. But there were still marked as invalid.
      Usually  this is a matter of duplicates: same IP address  with  different
      name, or same object name with different IP address.
      However,  the ones they were conflicting with (marked as  valid  records)
      were  processed with no further problems. If this was a processing  error
      of  any  kind,  you  can  always create  the  object  again  whithin  the
      Check Point SmartDashboard.
      Please run Object Filler again over the same source and use this time the
      "-v" (verbose) parameter. The  output  will  tell  you  in  the  object's
      processing line more details on why such objects failed.

===============================================================================
*** Successfully imported configuration from Cisco PIX!
-------------------------------------------------------------------------------
It took 3.0 seconds of total processing time on QUIET Mode.
Processed 362 possible objects and/or rules.
Found 39 total valid (or successfully processed) objects/rules.
Found 7 total NATted records.
-------------------------------------------------------------------------------
Total successfully processed Hosts = 20
Total successfully processed Networks = 7
Total successfully processed TCP Service objects = 6
Total successfully processed UDP Service objects = 6
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

The command line was the same with the difference of the “-a” switch which was substituted with the “-o” switch. As expected the output is more or less the same as well.

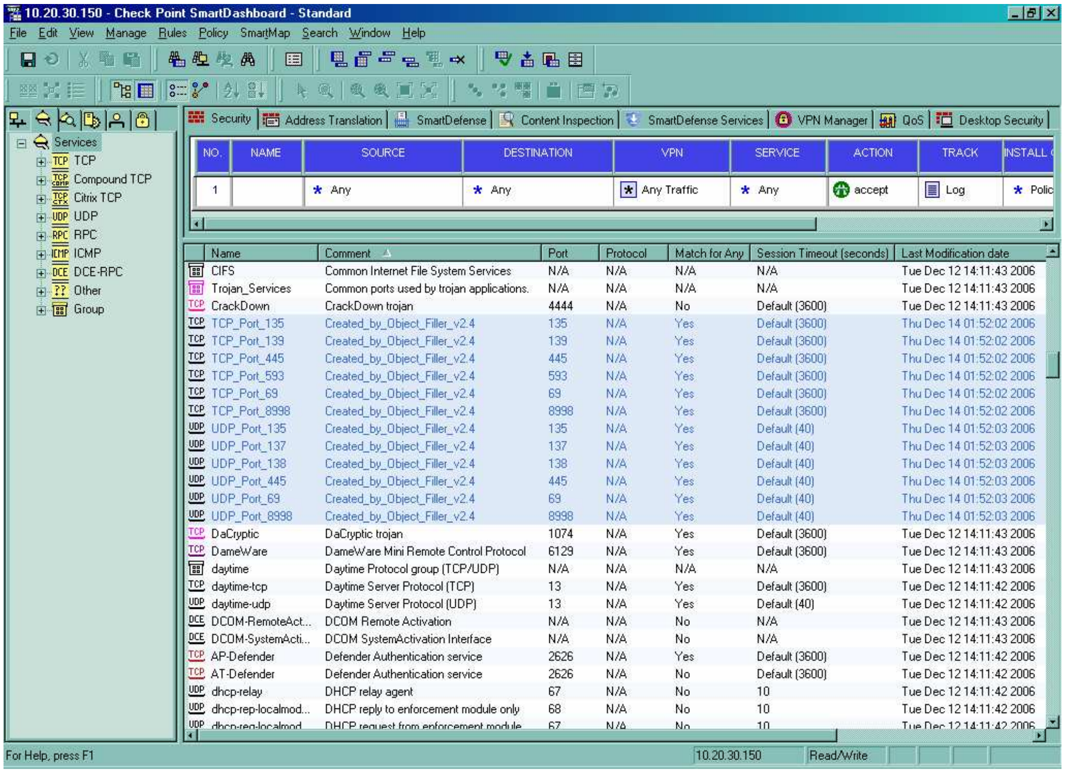
Let’s now import the objects. First as usual, you have to transfer them. By now, you should know how to do this, so I’ll save time and space by not showing that, but just showing the file is actually there:

```
[Expert@ngxr62]# ls -la mypix1.dbedit
-rw-rw----    1 root     root        12768 Nov 13 21:17 mypix1.dbedit
```

Now the critical step of importing the DBedit script.

```
[Expert@ngxr62]# dbedit -f mypix1.dbedit -s localhost -u admin
Enter Administrator Password: secret
Host_172.16.14.11 updated successfully.
Host_172.16.28.1 updated successfully.
    .
    .
    .
Host_10.20.229.12 updated successfully.
Host_10.20.229.13 updated successfully.
[Expert@ngxr62]#
```

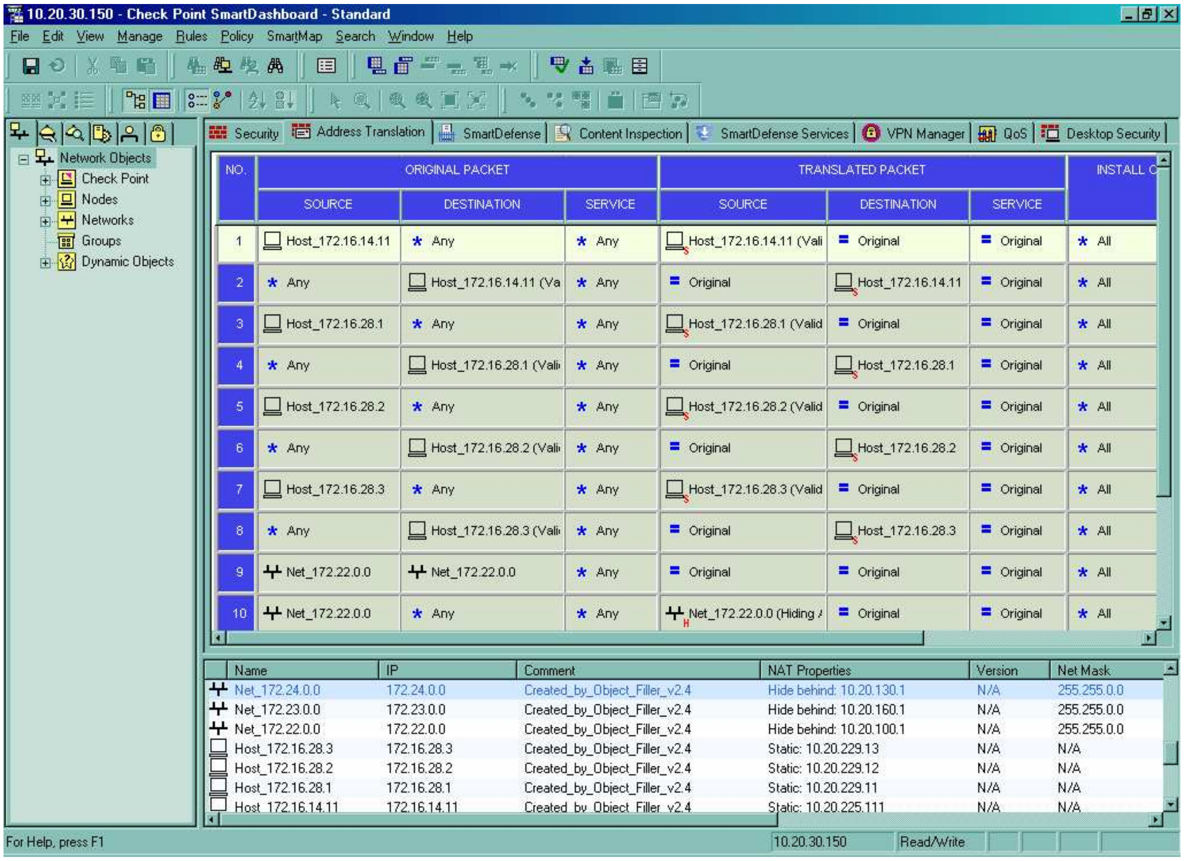
As usual, now the proof that we imported the objects. This time, we'll show the new services created, which would be the different thing from previous processings…



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL (
1		* Any	* Any	Any Traffic	* Any	accept	Log	* Polic

Name	Comment	Port	Protocol	Match for Any	Session Timeout (seconds)	Last Modification date
CIFS	Common Internet File System Services	N/A	N/A	N/A	N/A	Tue Dec 12 14:11:43 2006
Trojan_Services	Common ports used by trojan applications.	N/A	N/A	N/A	N/A	Tue Dec 12 14:11:43 2006
CrackDown	CrackDown trojan	4444	N/A	No	Default (3600)	Tue Dec 12 14:11:43 2006
TCP_Port_135	Created_by_Object_Filler_v2.4	135	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
TCP_Port_139	Created_by_Object_Filler_v2.4	139	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
TCP_Port_445	Created_by_Object_Filler_v2.4	445	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
TCP_Port_593	Created_by_Object_Filler_v2.4	593	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
TCP_Port_69	Created_by_Object_Filler_v2.4	69	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
TCP_Port_8998	Created_by_Object_Filler_v2.4	8998	N/A	Yes	Default (3600)	Thu Dec 14 01:52:02 2006
UDP_Port_135	Created_by_Object_Filler_v2.4	135	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
UDP_Port_137	Created_by_Object_Filler_v2.4	137	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
UDP_Port_138	Created_by_Object_Filler_v2.4	138	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
UDP_Port_445	Created_by_Object_Filler_v2.4	445	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
UDP_Port_69	Created_by_Object_Filler_v2.4	69	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
UDP_Port_8998	Created_by_Object_Filler_v2.4	8998	N/A	Yes	Default (40)	Thu Dec 14 01:52:03 2006
DaCryptic	DaCryptic trojan	1074	N/A	Yes	Default (3600)	Tue Dec 12 14:11:43 2006
DameWare	DameWare Mini Remote Control Protocol	6129	N/A	Yes	Default (3600)	Tue Dec 12 14:11:43 2006
daytime	Daytime Protocol group (TCP/UDP)	N/A	N/A	N/A	N/A	Tue Dec 12 14:11:43 2006
daytime-tcp	Daytime Server Protocol (TCP)	13	N/A	Yes	Default (3600)	Tue Dec 12 14:11:42 2006
daytime-udp	Daytime Server Protocol (UDP)	13	N/A	Yes	Default (40)	Tue Dec 12 14:11:42 2006
DCOM-RemoteAct...	DCOM Remote Activation	N/A	N/A	No	N/A	Tue Dec 12 14:11:43 2006
DCOM-SystemActi...	DCOM SystemActivation Interface	N/A	N/A	No	N/A	Tue Dec 12 14:11:43 2006
AP-Defender	Defender Authentication service	2626	N/A	Yes	Default (3600)	Tue Dec 12 14:11:42 2006
AT-Defender	Defender Authentication service	2626	N/A	No	Default (3600)	Tue Dec 12 14:11:42 2006
dhcp-relay	DHCP relay agent	67	N/A	No	10	Tue Dec 12 14:11:42 2006
dhcp-rep-localmod...	DHCP reply to enforcement module only	68	N/A	No	10	Tue Dec 12 14:11:42 2006
dhcp-req-localmod	DHCP request from enforcement module	67	N/A	No	10	Tue Dec 12 14:11:42 2006

You may take a look as well at the Address Translation rules as well. As the NATted objects are processed and its properties are set with Automatic NAT, rules are automatically generated, and they are shown in the proper screen:



10.20.30.150 - Check Point SmartDashboard - Standard

NO.	ORIGINAL PACKET			TRANSLATED PACKET			INSTALL O
	SOURCE	DESTINATION	SERVICE	SOURCE	DESTINATION	SERVICE	
1	Host_172.16.14.11	* Any	* Any	Host_172.16.14.11 (Vali	= Original	= Original	* All
2	* Any	Host_172.16.14.11 (Va	* Any	= Original	Host_172.16.14.11	= Original	* All
3	Host_172.16.28.1	* Any	* Any	Host_172.16.28.1 (Valid	= Original	= Original	* All
4	* Any	Host_172.16.28.1 (Vali	* Any	= Original	Host_172.16.28.1	= Original	* All
5	Host_172.16.28.2	* Any	* Any	Host_172.16.28.2 (Valid	= Original	= Original	* All
6	* Any	Host_172.16.28.2 (Vali	* Any	= Original	Host_172.16.28.2	= Original	* All
7	Host_172.16.28.3	* Any	* Any	Host_172.16.28.3 (Valid	= Original	= Original	* All
8	* Any	Host_172.16.28.3 (Vali	* Any	= Original	Host_172.16.28.3	= Original	* All
9	Net_172.22.0.0	Net_172.22.0.0	* Any	= Original	= Original	= Original	* All
10	Net_172.22.0.0	* Any	* Any	Net_172.22.0.0 (Hiding /	= Original	= Original	* All

Name	IP	Comment	NAT Properties	Version	Net Mask
Net_172.24.0.0	172.24.0.0	Created_by_Object_Filler_v2.4	Hide behind: 10.20.130.1	N/A	255.255.0.0
Net_172.23.0.0	172.23.0.0	Created_by_Object_Filler_v2.4	Hide behind: 10.20.160.1	N/A	255.255.0.0
Net_172.22.0.0	172.22.0.0	Created_by_Object_Filler_v2.4	Hide behind: 10.20.100.1	N/A	255.255.0.0
Host_172.16.28.3	172.16.28.3	Created_by_Object_Filler_v2.4	Static: 10.20.229.13	N/A	N/A
Host_172.16.28.2	172.16.28.2	Created_by_Object_Filler_v2.4	Static: 10.20.229.12	N/A	N/A
Host_172.16.28.1	172.16.28.1	Created_by_Object_Filler_v2.4	Static: 10.20.229.11	N/A	N/A
Host_172.16.14.11	172.16.14.11	Created_by_Object_Filler_v2.4	Static: 10.20.225.111	N/A	N/A

And you are done! – Congratulations!

As you saw, converting from a Cisco PIX configuration may not be that complicated after all. At least the objects… ;-)

5.4 Importing rules from Cisco PIX/ASA/FWSM and Cisco Routers configurations

Beautiful! Importing just objects from a Cisco PIX/ASA/FWSM configuration was a piece of cake. Now you are ready for something stronger…

The next logical step would be to import now rules. So, here you go…

The first thing you have to know, is that if you already imported your objects (i.e. if you went through the section 5.3) you are half way done. If you haven't imported your objects then don't worry, we'll show you in this section the whole thing anyway.

The second thing you need to know is that there are certain limitations when importing PIX configurations. This feature of importing rules works with many configurations, but has not been tested a lot to assure you it will work *always* – you may help: if you own sample PIX files that get you problems when importing, send them my way so I can work on fixing the problem.

First the things that will work: Importing *access-lists* will work. Importing *conduit* and *outbound* statements doesn't work. Cisco recommends using *access-lists* on the more recent PIX/ASA/FWSM configurations anyways, so most of the people is expected to have *access-lists* and this should not be a major problem.

Then, there's a problem: if you have groups (*service* or *network object-groups*) or *names* included in the *access-lists* then Object Filler 2.2 will miserably fail in such situations. You **must** use **Object Filler 2.4** or a more recent version to make this work – For deeper information please refer to the User's Manual. This is actually important, especially if you are converting a large number of rules where you can save (literally) hours of boring labor…

That being said, we'll assume the same *jenny_pix_rulebase*.txt file that we used on section 5.3, with exactly the same contents

```
D:\Stuff\OFiller\V2.4>dir jenny_pix_rulebase.txt
 Volume in drive D is Data
 Volume Serial Number is 88AE-1253

 Directory of D:\Stuff\OFiller\V2.4

13/11/2005  06:15 p.m.             2,364 jenny_pix_rulebase.txt
               1 File(s)          2,364 bytes
               0 Dir(s)     678,948,864 bytes free
```

Now, we'll run Object Filler over this file:

```
D:\Stuff\OFiller\V2.4>ofiller -f jenny_pix_rulebase.txt -i pix -a mypixrules.csv
 -p pixpol -nopv
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

Processing NATted Objects...
..
Processing Regular Objects...
...........

-------------------------------------------------------------------------------
Verifying existing records - creating unknown services
-------------------------------------------------------------------------------

Processing Rules...
................................................................................
..........
-------------------------------------------------------------------------------
```

```
Total invalid objects processed = 19
-------------------------------------------------------------------------------
NOTE: These invalid objects are the ones that were not processed  successfully.
      Unless you used the No Objects or Policy Verification switch (-nopv);  on
      which case, the objects were processed  normally  as  if  they  were  not
      conflicting with something else. But there were still marked as invalid.
      Usually  this is a matter of duplicates: same IP address  with  different
      name, or same object name with different IP address.
      However,  the ones they were conflicting with (marked as  valid  records)
      were  processed with no further problems. If this was a processing  error
      of  any  kind,  you  can  always create  the  object  again  whithin  the
      Check Point SmartDashboard.
      Please run Object Filler again over the same source and use this time the
      "-v" (verbose) parameter. The  output  will  tell  you  in  the  object's
      processing line more details on why such objects failed.

===============================================================================
*** Successfully imported configuration from Cisco PIX!
-------------------------------------------------------------------------------
It took 3.0 seconds of total processing time on QUIET Mode.
Processed 439 possible objects and/or rules.
Found 69 total valid (or successfully processed) objects/rules.
Found 7 total NATted records.
Rules processing was requested and done for Policy "pixpol".
-------------------------------------------------------------------------------
Total successfully processed Hosts = 20
Total successfully processed Networks = 7
Total successfully processed TCP Service objects = 6
Total successfully processed UDP Service objects = 6
Total successfully processed Rules = 30
-------------------------------------------------------------------------------
Please review that all CSV output information was written correctly.
Please remember CSV information is for reviewing only. For importing into
SmartCenter you need to use DBedit mode (-o switch).
===============================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

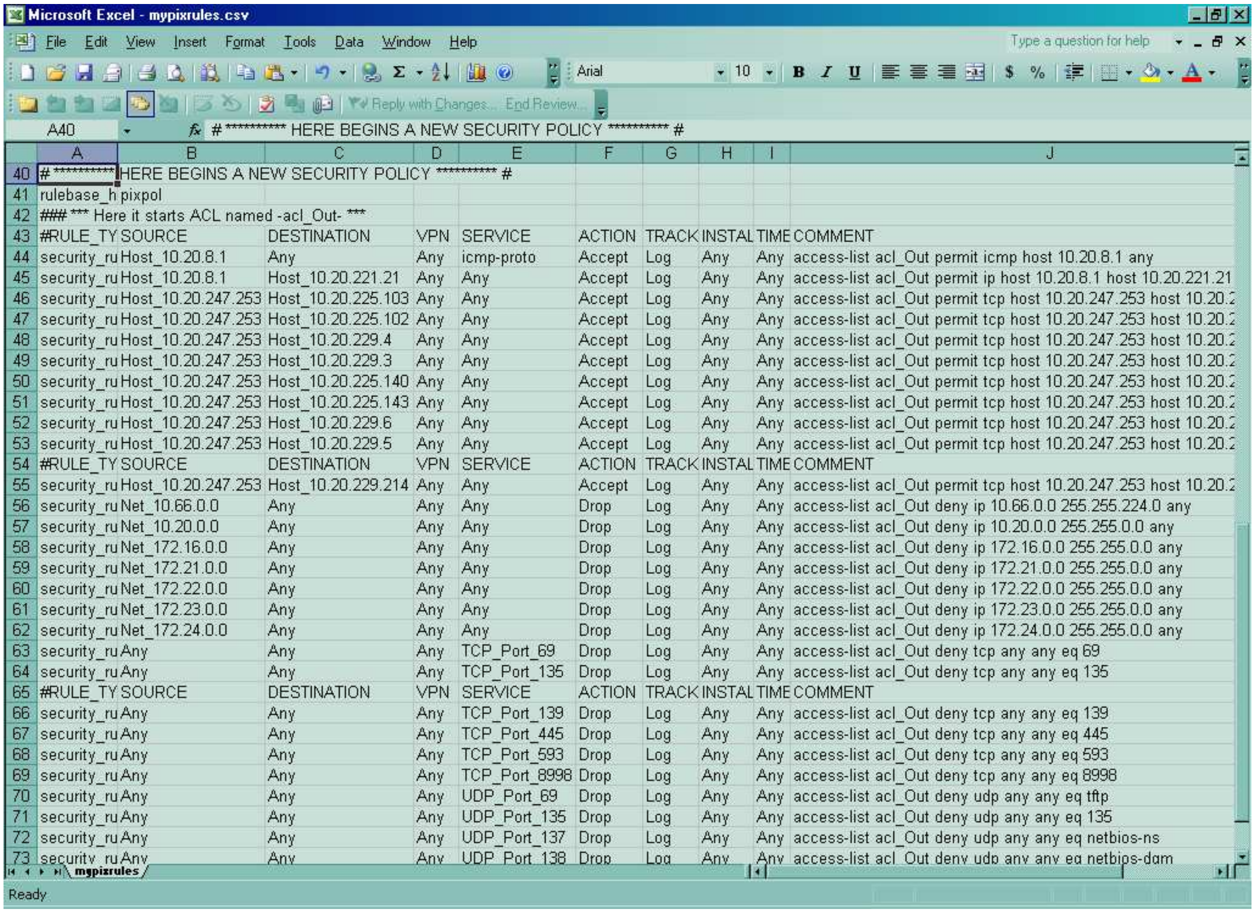
What in heaven happened here?
The basic objects importing stuff explanation was given in section 5.3, so we’ll skip to the new things relevant to rules.

When invoking Object Filler, this time we added two options to the command line:

`-p pixpol` This tells Object Filler to process rules and leave such rules in a policy named pixpol, which will be created whenever the DBedit commands are generated.

`-nopv` This tells Object Filler not to verify that the objects that are used in the rulebase were actually created or processed on this run. By default Object Filler verifies that all the objects (services and network objects) used in the rules actually exist. Since Object Filler can’t connect to the SmartCenter to verify that, it assumes that the objects were processed in the same run. Some times, if you are processing just rules, this may not be the case, so telling the tool NOT to perform this verification is actually useful. While in the specific case of this file we’re processing now was not needed, still we used it so we can show it to you ;-)

Ok. Let’s take a look at the resulting CSV file now:


Microsoft Excel - mypixrules.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
mypixrules
Ready

A40 # ********** HERE BEGINS A NEW SECURITY POLICY ********** #

	A	B	C	D	E	F	G	H	I	J
40	# ********** HERE BEGINS A NEW SECURITY POLICY ********** #									
41	rulebase_h	pixpol								
42	### *** Here it starts ACL named -acl_Out- ***									
43	#RULE_TY	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTAL	TIME	COMMENT
44	security_ru	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	Any	Any	access-list acl_Out permit icmp host 10.20.8.1 any
45	security_ru	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.221.21
46	security_ru	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
47	security_ru	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
48	security_ru	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
49	security_ru	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
50	security_ru	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
51	security_ru	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
52	security_ru	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
53	security_ru	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
54	#RULE_TY	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTAL	TIME	COMMENT
55	security_ru	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Log	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.2
56	security_ru	Net_10.66.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
57	security_ru	Net_10.20.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 10.20.0.0 255.255.0.0 any
58	security_ru	Net_172.16.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 172.16.0.0 255.255.0.0 any
59	security_ru	Net_172.21.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 172.21.0.0 255.255.0.0 any
60	security_ru	Net_172.22.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 172.22.0.0 255.255.0.0 any
61	security_ru	Net_172.23.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 172.23.0.0 255.255.0.0 any
62	security_ru	Net_172.24.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 172.24.0.0 255.255.0.0 any
63	security_ru	Any	Any	Any	TCP_Port_69	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 69
64	security_ru	Any	Any	Any	TCP_Port_135	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 135
65	#RULE_TY	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTAL	TIME	COMMENT
66	security_ru	Any	Any	Any	TCP_Port_139	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 139
67	security_ru	Any	Any	Any	TCP_Port_445	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 445
68	security_ru	Any	Any	Any	TCP_Port_593	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 593
69	security_ru	Any	Any	Any	TCP_Port_8998	Drop	Log	Any	Any	access-list acl_Out deny tcp any any eq 8998
70	security_ru	Any	Any	Any	UDP_Port_69	Drop	Log	Any	Any	access-list acl_Out deny udp any any eq tftp
71	security_ru	Any	Any	Any	UDP_Port_135	Drop	Log	Any	Any	access-list acl_Out deny udp any any eq 135
72	security_ru	Any	Any	Any	UDP_Port_137	Drop	Log	Any	Any	access-list acl_Out deny udp any any eq netbios-ns
73	security_ru	Any	Any	Any	UDP_Port_138	Drop	Log	Any	Any	access-list acl_Out deny udp any any eq netbios-dgm

Wow! Doesn’t that look beautiful?

Let’s explain several interesting things:

a) The file contains actually objects and rules, not only rules. The objects are shown at the beginning and the rules are left at the bottom of the file. You may actually see both there. The screenshot above just shows the Rules Section.
b) The Rules Section begins with a comment that says *“Here begins the Rule Section”*. I think that’s quite clear.
c) Then, the *rulebase*_header keyword in the first column indicates there it begins a new Policy. Let’s remember that a policy is a set of rules.
d) Note that all the rows that represent rules use the *security*_rule keyword in the first column. Then the following columns are the same columns that you see on SmartDashboard when you use Simplified Mode. By default the *Track* property is set to log, the *Install On* and *Time* columns are set to *Any*.

e) All the translated rules appear with the *Comment* Column showing the original rule that originated the rule of such row. This can actually be used to audit and review that the rules were accurately translated. If you see something strange, don't hesitate and run Object Filler again using the verbose mode (`-v`) to get more information on what actually happened. This is important, so you avoid problems later.

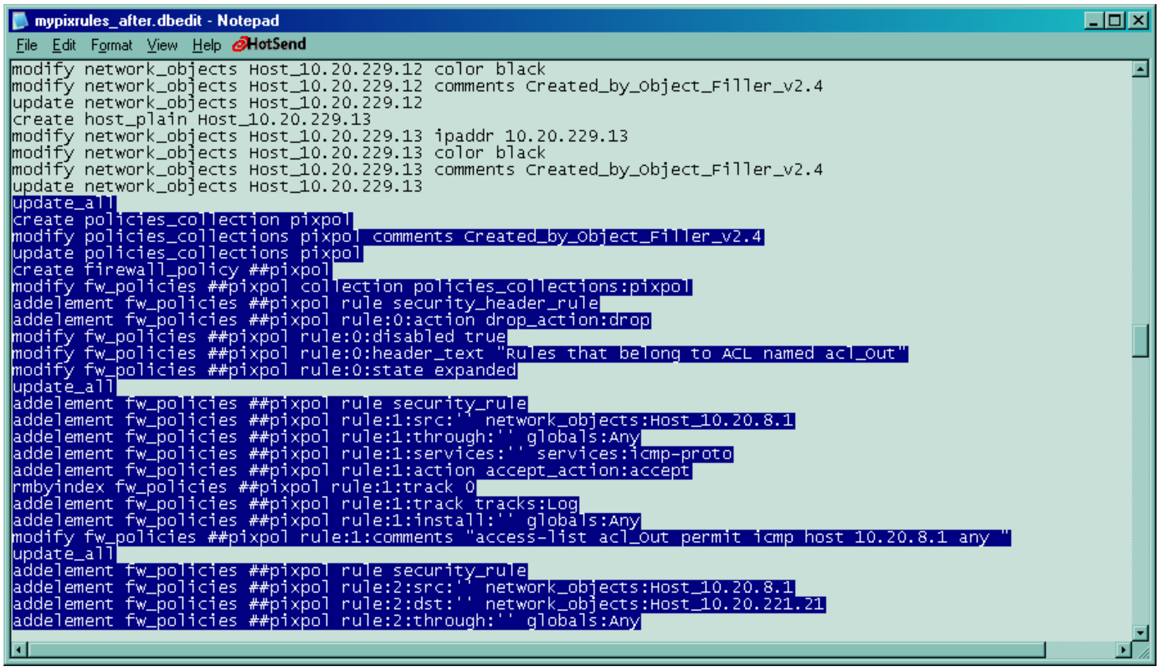
f) Mhh… thinking it twice… one ***strongly*** suggested thing is to use NOW (yes, NOW ☺) the –*v* switch just as an exercise. Redirect the output to a file and then view it with a text editor. Get familiar with this report as it gives you details on how the processing is done, line by line. The syntax you should use would be **`ofiller -f jenny_pix_rulebase.txt -i pix -a mypixrules.csv -p pixpol –nopv > output.txt`** - and then review the **`output.txt`** file – if you already ran the command once, remember to change the name of the output file (**`mypixrules.csv`** on this case) as Object Filler never overwrites a file by itself, to prevent deleting important information by accident.

Once you review the Spreadsheet and you are sure all is right, then you may go to the next step and produce the DBedit commands:

```
D:\Stuff\OFiller\V2.4>ofiller -f jenny_pix_rulebase.txt -i pix -o mypixrules.dbe
dit -p pixpol -nopv
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

Processing NATted Objects...
..
Processing Regular Objects...
...........

-------------------------------------------------------------------------------
Verifying existing records - creating unknown services
-------------------------------------------------------------------------------

Processing Rules...
................................................................................
..........
-------------------------------------------------------------------------------
Total invalid objects processed = 19
-------------------------------------------------------------------------------
NOTE: These invalid objects are the ones that were not processed  successfully.
      Unless you used the No Objects or Policy Verification switch (-nopv);  on
      which case, the objects were processed  normally  as  if  they  were  not
      conflicting with something else. But there were still marked as invalid.
      Usually  this is a matter of duplicates: same IP address  with  different
      name, or same object name with different IP address.
      However,  the ones they were conflicting with (marked as  valid  records)
      were  processed with no further problems. If this was a processing  error
      of  any  kind,  you  can  always create  the  object  again  whithin  the
      Check Point SmartDashboard.
      Please run Object Filler again over the same source and use this time the
      "-v" (verbose) parameter. The  output  will  tell  you  in  the  object's
      processing line more details on why such objects failed.
```

```
================================================================================
*** Successfully imported configuration from Cisco PIX!
--------------------------------------------------------------------------------
It took 3.0 seconds of total processing time on QUIET Mode.
Processed 439 possible objects and/or rules.
Found 69 total valid (or successfully processed) objects/rules.
Found 7 total NATted records.
Rules processing was requested and done for Policy "pixpol".
--------------------------------------------------------------------------------
Total successfully processed Hosts = 20
Total successfully processed Networks = 7
Total successfully processed TCP Service objects = 6
Total successfully processed UDP Service objects = 6
Total successfully processed Rules = 31
--------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
================================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

Now you are ready to import those to the SmartCenter, right? – Well, there is one small detail to take care of first. Small, but quite important: if you already have the objects imported into your SmartCenter –as it may be the case if you did the exercise in section 5.3- then you need to delete the DBedit commands that create such objects. So, let’s edit the file and see…



mypixrules_after.dbedit - Notepad
File Edit Format View Help HotSend

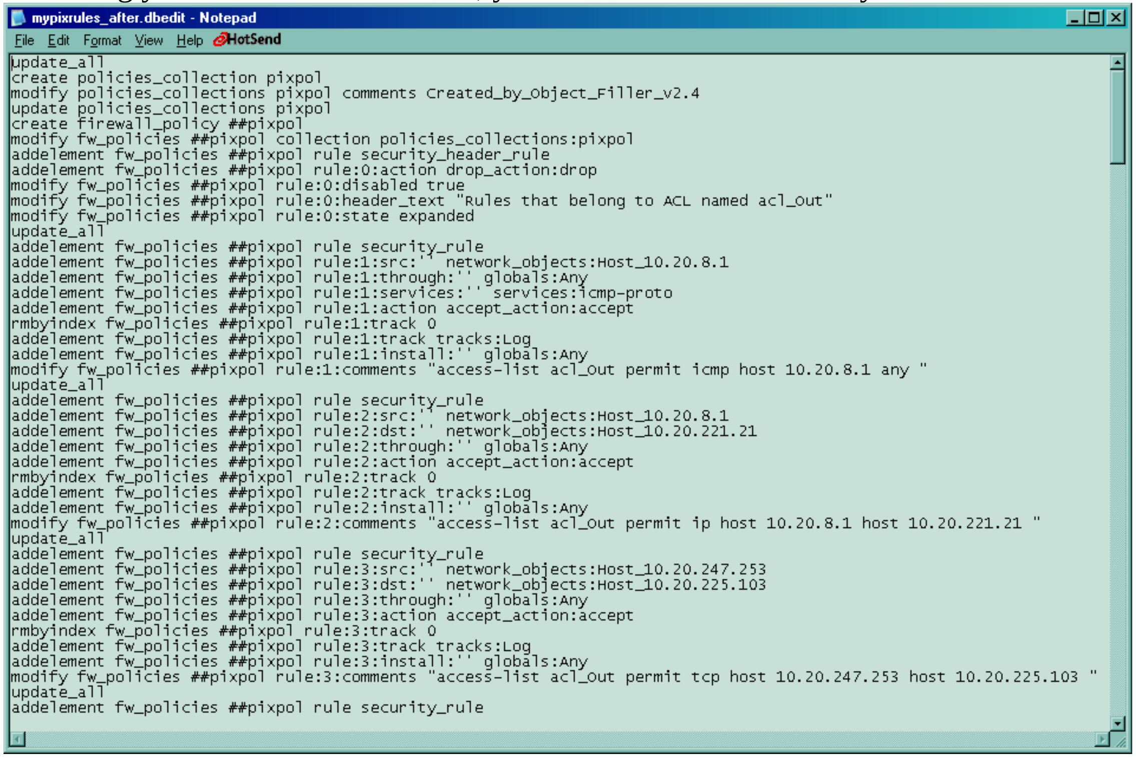
```
modify network_objects Host_10.20.229.12 color black
modify network_objects Host_10.20.229.12 comments Created_by_Object_Filler_v2.4
update network_objects Host_10.20.229.12
create host_plain Host_10.20.229.13
modify network_objects Host_10.20.229.13 ipaddr 10.20.229.13
modify network_objects Host_10.20.229.13 color black
modify network_objects Host_10.20.229.13 comments Created_by_Object_Filler_v2.4
update network_objects Host_10.20.229.13
update_all
create policies_collection pixpol
modify policies_collections pixpol comments Created_by_Object_Filler_v2.4
update policies_collections pixpol
create firewall_policy ##pixpol
modify fw_policies ##pixpol collection policies_collections:pixpol
addelement fw_policies ##pixpol rule security_header_rule
addelement fw_policies ##pixpol rule:0:action drop_action:drop
modify fw_policies ##pixpol rule:0:disabled true
modify fw_policies ##pixpol rule:0:header_text "Rules that belong to ACL named acl_out"
modify fw_policies ##pixpol rule:0:state expanded
update_all
addelement fw_policies ##pixpol rule security_rule
addelement fw_policies ##pixpol rule:1:src:'' network_objects:Host_10.20.8.1
addelement fw_policies ##pixpol rule:1:through:'' globals:Any
addelement fw_policies ##pixpol rule:1:services:'' services:icmp-proto
addelement fw_policies ##pixpol rule:1:action accept_action:accept
rmbyindex fw_policies ##pixpol rule:1:track 0
addelement fw_policies ##pixpol rule:1:track tracks:Log
addelement fw_policies ##pixpol rule:1:install:'' globals:Any
modify fw_policies ##pixpol rule:1:comments "access-list acl_out permit icmp host 10.20.8.1 any "
update_all
addelement fw_policies ##pixpol rule security_rule
addelement fw_policies ##pixpol rule:2:src:'' network_objects:Host_10.20.8.1
addelement fw_policies ##pixpol rule:2:dst:'' network_objects:Host_10.20.221.21
addelement fw_policies ##pixpol rule:2:through:'' globals:Any
```

There is a point in the file where an *update_all* DBedit command is used, and then a *create policies_collection* is seen. That is the place where the rules creation begins.

So, let me repeat. If you already have the objects created, delete all the lines **before** the *update_all* line, or you will get plenty of ugly errors. If you delete the initial lines, don't leave any blank line at the beginning of the file or you'll get errors as well

If you did not create the objects before, you can go ahead and use the file as it is at this moment.

Assuming you had to delete the lines, your file should look this way now:


mypixrules_after.dbedit - Notepad
File Edit Format View Help HotSend

```
update_all
create policies_collection pixpol
modify policies_collections pixpol comments Created_by_Object_Filler_v2.4
update policies_collections pixpol
create firewall_policy ##pixpol
modify fw_policies ##pixpol collection policies_collections:pixpol
addelement fw_policies ##pixpol rule security_header_rule
addelement fw_policies ##pixpol rule:0:action drop_action:drop
modify fw_policies ##pixpol rule:0:disabled true
modify fw_policies ##pixpol rule:0:header_text "Rules that belong to ACL named acl_out"
modify fw_policies ##pixpol rule:0:state expanded
update_all
addelement fw_policies ##pixpol rule security_rule
addelement fw_policies ##pixpol rule:1:src:'' network_objects:Host_10.20.8.1
addelement fw_policies ##pixpol rule:1:through:'' globals:Any
addelement fw_policies ##pixpol rule:1:services:'' services:icmp-proto
addelement fw_policies ##pixpol rule:1:action accept_action:accept
rmbyindex fw_policies ##pixpol rule:1:track 0
addelement fw_policies ##pixpol rule:1:track tracks:Log
addelement fw_policies ##pixpol rule:1:install:'' globals:Any
modify fw_policies ##pixpol rule:1:comments "access-list acl_out permit icmp host 10.20.8.1 any "
update_all
addelement fw_policies ##pixpol rule security_rule
addelement fw_policies ##pixpol rule:2:src:'' network_objects:Host_10.20.8.1
addelement fw_policies ##pixpol rule:2:dst:'' network_objects:Host_10.20.221.21
addelement fw_policies ##pixpol rule:2:through:'' globals:Any
addelement fw_policies ##pixpol rule:2:action accept_action:accept
rmbyindex fw_policies ##pixpol rule:2:track 0
addelement fw_policies ##pixpol rule:2:track tracks:Log
addelement fw_policies ##pixpol rule:2:install:'' globals:Any
modify fw_policies ##pixpol rule:2:comments "access-list acl_out permit ip host 10.20.8.1 host 10.20.221.21 "
update_all
addelement fw_policies ##pixpol rule security_rule
addelement fw_policies ##pixpol rule:3:src:'' network_objects:Host_10.20.247.253
addelement fw_policies ##pixpol rule:3:dst:'' network_objects:Host_10.20.225.103
addelement fw_policies ##pixpol rule:3:through:'' globals:Any
addelement fw_policies ##pixpol rule:3:action accept_action:accept
rmbyindex fw_policies ##pixpol rule:3:track 0
addelement fw_policies ##pixpol rule:3:track tracks:Log
addelement fw_policies ##pixpol rule:3:install:'' globals:Any
modify fw_policies ##pixpol rule:3:comments "access-list acl_out permit tcp host 10.20.247.253 host 10.20.225.103 "
update_all
addelement fw_policies ##pixpol rule security_rule
```

One last thing: if you are going to import this into an NG+AI R55W or below SmartCenter, you need to change some of the DBedit commands to avoid problems with earlier versions of DBedit. If you want to avoid problems, change all the lines that say

`addelement fw_policies ##`*`Policy_Name`*` rule:`*`N`*`:through:'' `*`something_here`*

and replace them with

`addelement fw_policies ##`*`Policy_Name`*` rule:`*`N`*`:through `*`something_here`*

I repeat. The above step is only necessary if you want to import rules into NG+AI R55W or below. This includes NG+AI R55, R54 and FP3.

The difference is in the colon and the apostrophes after *through*. In any case, when you are importing PIX configurations, you may safely ignore all this. Question 4.6 in the FAQ Section of the User's Manual documents this as well…

Great. Now let's continue our quest to conquer the world ;-)

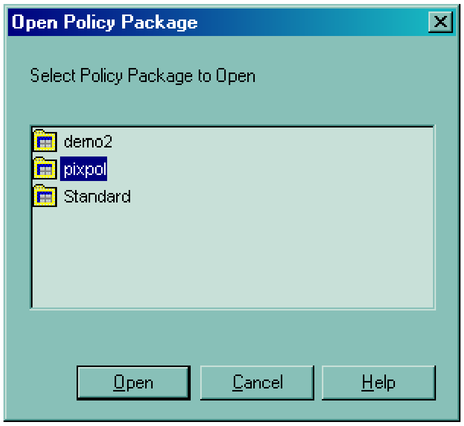
Pass the DBedit commands file to the SmartCenter, and then import it using DBedit as you would usually do, and the way is shown in the first sections of this document.

```
[Expert@ngxr62]# ls -la mypixrules_after.dbedit
-rw-rw----    1 root     root        17530 Dec 14 02:31 mypixrules_after.dbedit
[Expert@ngxr62]# dbedit -s localhost -u admin -p secret -f mypixrules_after.dbedit
Object_Filler_Imported_pixpol updated successfully.
fw_policies::##pixpol Updated Successfully
fw_policies::##pixpol Updated Successfully
     .
     .
     .
fw_policies::##pixpol Updated Successfully
fw_policies::##pixpol Updated Successfully
[Expert@ngxr62]#
```

Excellent! Great! At this point you already have imported the rules. Notice that this time we used the "`-p`" switch to tell DBedit the administrator's password directly in the command line…

Login to your SmartCenter using SmartDashboard. And you will see nothing there…

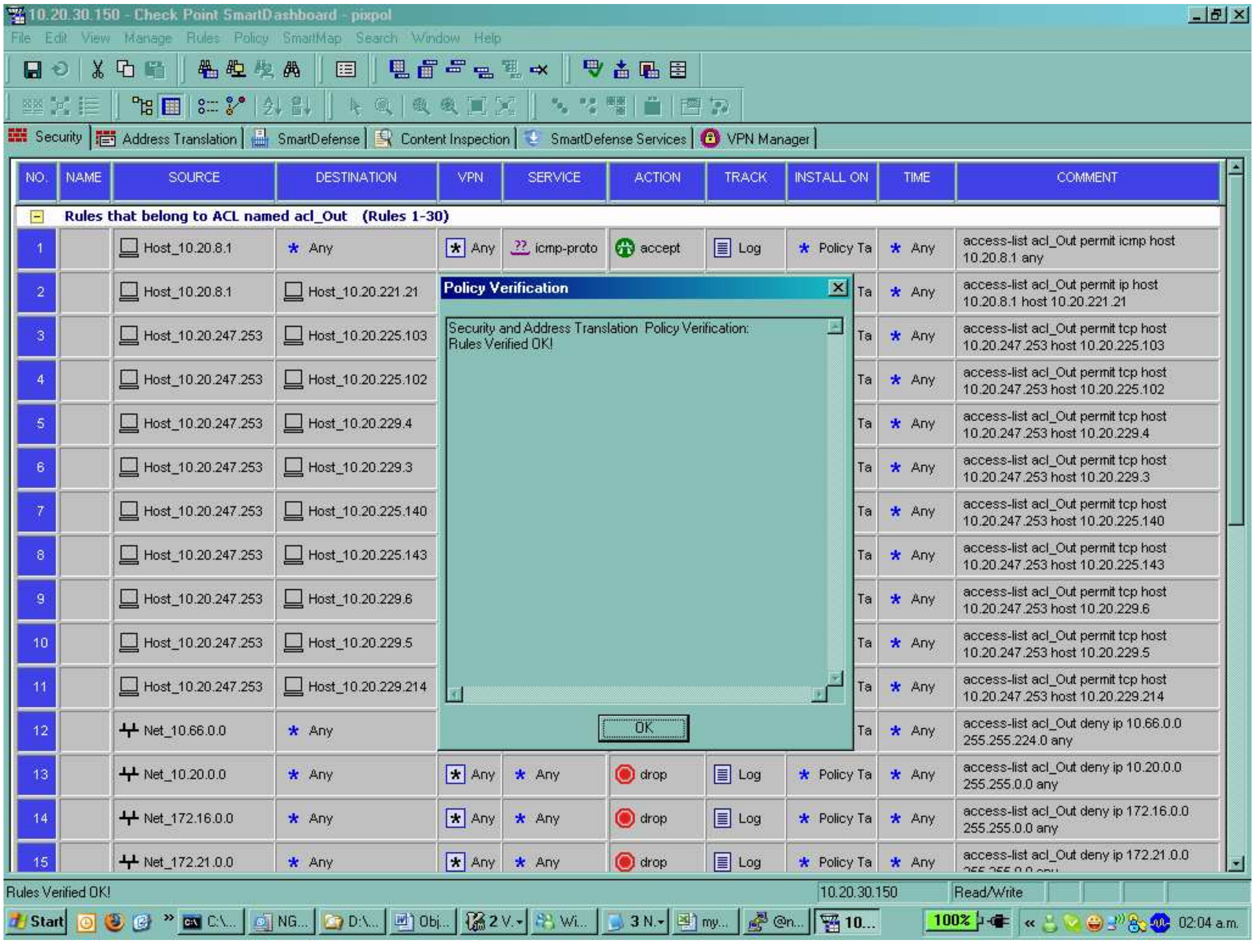
The reason is because the rules are kept in a separate policy package. To get them, go to the File Menu, select the Open option and you will see several policy packages and among them one (or perhaps several if this is not the first policy you import with Object Filler) named with the name you used with the *–p* switch when you invoked Object Filler. In this example, *pixpol*.



Choose then the imported policy, and *voila*! – you have your rules there… shining and waiting for you to apply them (after a backup and a second review, of course).

Nice! Did you like it? ☺

Finally, don't forget to verify the rulebase, just to make sure that there's nothing conflicting over there…



Rules that belong to ACL named acl_Out (Rules 1-30)

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1		Host_10.20.8.1	* Any	Any	icmp-proto	accept	Log	* Policy Ta	* Any	access-list acl_Out permit icmp host 10.20.8.1 any
2		Host_10.20.8.1	Host_10.20.221.21					Ta	* Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.221.21
3		Host_10.20.247.253	Host_10.20.225.103					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.103
4		Host_10.20.247.253	Host_10.20.225.102					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.102
5		Host_10.20.247.253	Host_10.20.229.4					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.4
6		Host_10.20.247.253	Host_10.20.229.3					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.3
7		Host_10.20.247.253	Host_10.20.225.140					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.140
8		Host_10.20.247.253	Host_10.20.225.143					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.143
9		Host_10.20.247.253	Host_10.20.229.6					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.6
10		Host_10.20.247.253	Host_10.20.229.5					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.5
11		Host_10.20.247.253	Host_10.20.229.214					Ta	* Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.214
12		Net_10.66.0.0	* Any					Ta	* Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
13		Net_10.20.0.0	* Any	Any	* Any	drop	Log	* Policy Ta	* Any	access-list acl_Out deny ip 10.20.0.0 255.255.0.0 any
14		Net_172.16.0.0	* Any	Any	* Any	drop	Log	* Policy Ta	* Any	access-list acl_Out deny ip 172.16.0.0 255.255.0.0 any
15		Net_172.21.0.0	* Any	Any	* Any	drop	Log	* Policy Ta	* Any	access-list acl_Out deny ip 172.21.0.0

Policy Verification

Security and Address Translation Policy Verification:
Rules Verified OK!

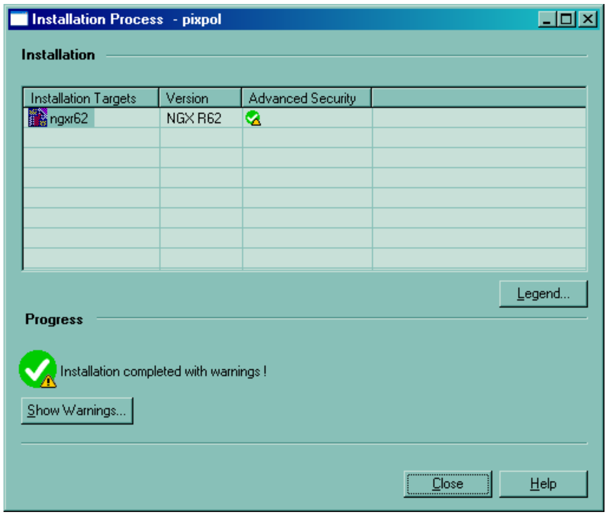
Rules Verified OK! 10.20.30.150 Read/Write

Most of the times, when you are converting from a PIX/ASA/FWSM policy, you will experience a bunch of errors here. Basically because one configuration normally contains several *access-lists*, which are separated here by rulebase tags, indicating you are indeed dealing with different policies (one access-list group can be considered as a policy by itself) inside the *master* security policy you imported.

You must remember that those Cisco products have the concept of applying the access-list per interface, and then there is a concept of security levels for each interface as well. Check Point VPN-1 (Pro/Power/UTM/Express) behaves more like a true router with security, thus making it easier to build a security policy (makes it easier to maintain as well), and some of the complexities of Cisco access-lists can be avoided.

It is strongly recommended if you see a bunch of errors here (and even if you don't get them) to go through the security policy and audit it to make sure the rules still apply to your new environment.

Ok. Let’s assume you manage to get the policy verified successfully (perhaps after you made some adjustments). Now let’s install the Policy



Installation Process - pixpol

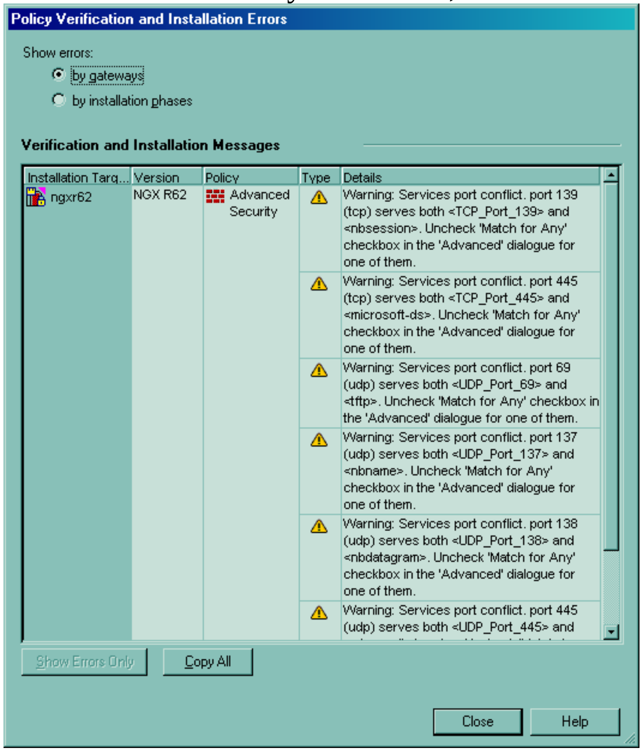
Installation

Installation Targets	Version	Advanced Security
ngxr62	NGX R62	✓

Progress

Installation completed with warnings !

You will see the Policy is installed, but some warning will be shown. Let’s see why:



Policy Verification and Installation Errors

Show errors:

● by gateways
○ by installation phases

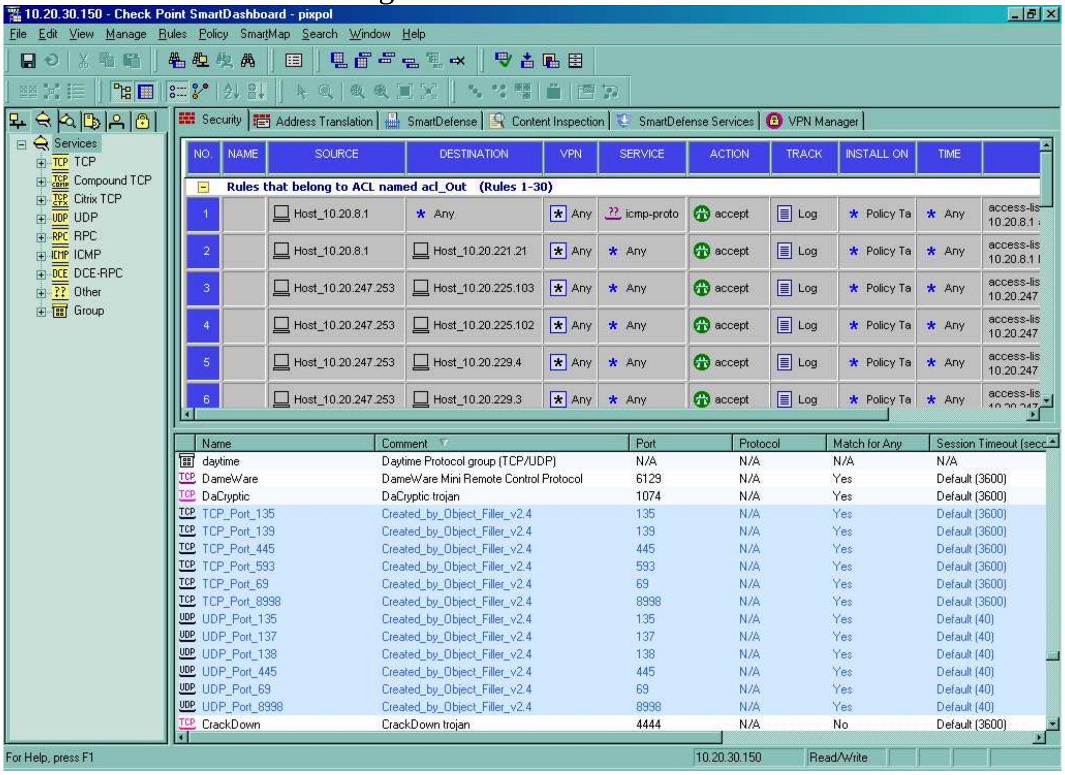
Verification and Installation Messages

Installation Targ...	Version	Policy	Type	Details
ngxr62	NGX R62	Advanced Security	⚠	Warning: Services port conflict. port 139 (tcp) serves both <TCP_Port_139> and <nbsession>. Uncheck 'Match for Any' checkbox in the 'Advanced' dialogue for one of them.
			⚠	Warning: Services port conflict. port 445 (tcp) serves both <TCP_Port_445> and <microsoft-ds>. Uncheck 'Match for Any' checkbox in the 'Advanced' dialogue for one of them.
			⚠	Warning: Services port conflict. port 69 (udp) serves both <UDP_Port_69> and <tftp>. Uncheck 'Match for Any' checkbox in the 'Advanced' dialogue for one of them.
			⚠	Warning: Services port conflict. port 137 (udp) serves both <UDP_Port_137> and <nbname>. Uncheck 'Match for Any' checkbox in the 'Advanced' dialogue for one of them.
			⚠	Warning: Services port conflict. port 138 (udp) serves both <UDP_Port_138> and <nbdatagram>. Uncheck 'Match for Any' checkbox in the 'Advanced' dialogue for one of them.
			⚠	Warning: Services port conflict. port 445 (udp) serves both <UDP_Port_445> and

Oh! Right! Object Filler by default sets the “Match for Any” property in all the new created services. If you don’t like this behavior… Well… then you either remove the commands from the DBedit file, or manually disable that from the services.

You may easily identify what services were created by Object Filler, because all of them have as a comment *"Created_by_Object_Filler_vn.n"*, and they begin with the *TCP_* or *UDP_* prefix, followed by the port number they represent…

This is shown in the following screenshot:



Rules that belong to ACL named acl_Out (Rules 1-30)

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	
1		Host_10.20.8.1	Any	Any	icmp-proto	accept	Log	Policy Ta	Any	access-lis 10.20.8.1
2		Host_10.20.8.1	Host_10.20.221.21	Any	Any	accept	Log	Policy Ta	Any	access-lis 10.20.8.1
3		Host_10.20.247.253	Host_10.20.225.103	Any	Any	accept	Log	Policy Ta	Any	access-lis 10.20.247
4		Host_10.20.247.253	Host_10.20.225.102	Any	Any	accept	Log	Policy Ta	Any	access-lis 10.20.247
5		Host_10.20.247.253	Host_10.20.229.4	Any	Any	accept	Log	Policy Ta	Any	access-lis 10.20.247
6		Host_10.20.247.253	Host_10.20.229.3	Any	Any	accept	Log	Policy Ta	Any	access-lis

Name	Comment	Port	Protocol	Match for Any	Session Timeout (secc
daytime	Daytime Protocol group (TCP/UDP)	N/A	N/A	N/A	N/A
DameWare	DameWare Mini Remote Control Protocol	6129	N/A	Yes	Default (3600)
DaCryptic	DaCryptic trojan	1074	N/A	Yes	Default (3600)
TCP_Port_135	Created_by_Object_Filler_v2.4	135	N/A	Yes	Default (3600)
TCP_Port_139	Created_by_Object_Filler_v2.4	139	N/A	Yes	Default (3600)
TCP_Port_445	Created_by_Object_Filler_v2.4	445	N/A	Yes	Default (3600)
TCP_Port_593	Created_by_Object_Filler_v2.4	593	N/A	Yes	Default (3600)
TCP_Port_69	Created_by_Object_Filler_v2.4	69	N/A	Yes	Default (3600)
TCP_Port_8998	Created_by_Object_Filler_v2.4	8998	N/A	Yes	Default (3600)
UDP_Port_135	Created_by_Object_Filler_v2.4	135	N/A	Yes	Default (40)
UDP_Port_137	Created_by_Object_Filler_v2.4	137	N/A	Yes	Default (40)
UDP_Port_138	Created_by_Object_Filler_v2.4	138	N/A	Yes	Default (40)
UDP_Port_445	Created_by_Object_Filler_v2.4	445	N/A	Yes	Default (40)
UDP_Port_69	Created_by_Object_Filler_v2.4	69	N/A	Yes	Default (40)
UDP_Port_8998	Created_by_Object_Filler_v2.4	8998	N/A	Yes	Default (40)
CrackDown	CrackDown trojan	4444	N/A	No	Default (3600)

Awesome!

So, want to know more? Perhaps you want to know how can you do this with other brands?

Well, well, the Manual has more information in it. It's highly recommended that you read it! – perhaps In that way you may find out some other interesting things … ☺ - Some day we may include more about importing files from other brands if you vote for it. So, call your local favorite radio station and give your vote… ;-)

6. Object Dumper

Object Dumper does the opposite thing than Object Filler. Instead of importing new objects into SmartCenter (or CMA in a Provider-1 environment) it actually exports what is already there, focusing on network and service objects mainly.

To do this, it parses the *Objects_5_0.C* file, the one found at *$FWDIR/conf* and about which we already discussed some things at the beginning of this document.

Be careful: While Object Filler can export the objects (and rules, as described in the User's manual), it should NOT be seen as a 100% efficient backup tool, as it is not. The reasons of why are explained in the User's Manual – but trust me, if you *only* use Object Dumper for backups, I don't want to be on your shoes, as sooner or later you will be in serious problems… 😐 - use other things (such as *upgrade_export* and *upgrade_import*) for backups instead.

6.1 Exporting network and service objects to a CSV File

That's the easiest thing you have done so far.

First, we have to transfer the *Objects_5_0.C* to the Windows machine where Object Dumper resides.

Remember that if you are going to do this in a Provider-1 installation, you have first to set the environment to the proper CMA, using the command `mdsenv` *`CMA_name`* – if you don't know what this means, don't worry – then you don't need this stuff. ☺

```
[Expert@ngxr60]# pwd
/home/admin
[Expert@ngxr60]# cd $FWDIR/conf
[Expert@ngxr60]# ls -la objects_5_0.C
-rw-rw----    1 root     root      1009999 Nov 13 21:20 objects_5_0.C
[Expert@ngxr60]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> put objects_5_0.C
local: objects_5_0.C remote: objects_5_0.C
227 Entering passive mode (10,20,30,76,4,105)
125 Using existing data connection
```

```
################################################################################
####################################################################
226 Closing data connection; File transfer successful.
1052119 bytes sent in 0.353 secs (2.9e+03 Kbytes/sec)
ftp> bye
221 Service closing control connection
```

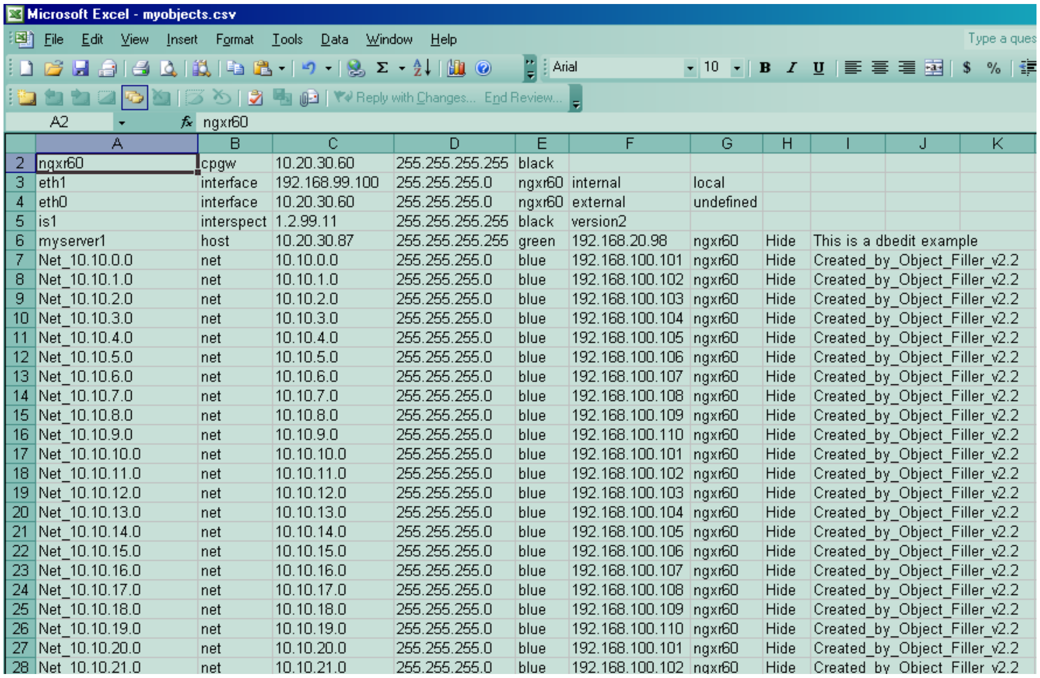
Now in your Windows machine, make sure the file is there

```
D:\Stuff\OFiller\V2.2>dir objects_5_0.C
 Volume in drive D is Data
 Volume Serial Number is 88AE-1253

 Directory of D:\Stuff\OFiller\V2.2

13/11/2005  08:44 p.m.         1,052,119 objects_5_0.C
               1 File(s)      1,052,119 bytes
               0 Dir(s)     760,270,848 bytes free
```

Ok. It’s there. It’s a fact. Now let’s use it as source to feed Object Dumper.

```
D:\Stuff\OFiller\V2.2>odumper -f objects_5_0.C -o myobjects.csv
Unofficial/Unsupported Object Dumper V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
Processing objects...
................................................................................
................................................................................
....................................................
===============================================================================
Processed 42120 possible objects and found 500 valid ones.
From these, 264 were NATted records. It took 3 seconds on quiet mode.
Total successfully processed CP Gateways = 1
Total successfully processed Hosts = 172
Total successfully processed InterSpect Devices = 1
Total successfully processed Networks = 266
Total successfully processed Interfaces = 2
Total successfully processed TCP Services = 6
Total successfully processed UDP Services = 6
Task done successfully! - Thank you for using Object Dumper V2.2!
```

Beautiful. We now have the objects exported. Let’s take a look on how they look like:


Microsoft Excel - myobjects.csv
File Edit View Insert Format Tools Data Window Help
Type a ques
Arial
10
Reply with Changes... End Review...
A2
ngxr60

	A	B	C	D	E	F	G	H	I	J	K
2	ngxr60	cpgw	10.20.30.60	255.255.255.255	black						
3	eth1	interface	192.168.99.100	255.255.255.0	ngxr60	internal	local				
4	eth0	interface	10.20.30.60	255.255.255.0	ngxr60	external	undefined				
5	is1	interspect	1.2.99.11	255.255.255.255	black	version2					
6	myserver1	host	10.20.30.87	255.255.255.255	green	192.168.20.98	ngxr60	Hide	This is a dbedit example		
7	Net_10.10.0.0	net	10.10.0.0	255.255.255.0	blue	192.168.100.101	ngxr60	Hide	Created_by_Object_Filler_v2.2		
8	Net_10.10.1.0	net	10.10.1.0	255.255.255.0	blue	192.168.100.102	ngxr60	Hide	Created_by_Object_Filler_v2.2		
9	Net_10.10.2.0	net	10.10.2.0	255.255.255.0	blue	192.168.100.103	ngxr60	Hide	Created_by_Object_Filler_v2.2		
10	Net_10.10.3.0	net	10.10.3.0	255.255.255.0	blue	192.168.100.104	ngxr60	Hide	Created_by_Object_Filler_v2.2		
11	Net_10.10.4.0	net	10.10.4.0	255.255.255.0	blue	192.168.100.105	ngxr60	Hide	Created_by_Object_Filler_v2.2		
12	Net_10.10.5.0	net	10.10.5.0	255.255.255.0	blue	192.168.100.106	ngxr60	Hide	Created_by_Object_Filler_v2.2		
13	Net_10.10.6.0	net	10.10.6.0	255.255.255.0	blue	192.168.100.107	ngxr60	Hide	Created_by_Object_Filler_v2.2		
14	Net_10.10.7.0	net	10.10.7.0	255.255.255.0	blue	192.168.100.108	ngxr60	Hide	Created_by_Object_Filler_v2.2		
15	Net_10.10.8.0	net	10.10.8.0	255.255.255.0	blue	192.168.100.109	ngxr60	Hide	Created_by_Object_Filler_v2.2		
16	Net_10.10.9.0	net	10.10.9.0	255.255.255.0	blue	192.168.100.110	ngxr60	Hide	Created_by_Object_Filler_v2.2		
17	Net_10.10.10.0	net	10.10.10.0	255.255.255.0	blue	192.168.100.101	ngxr60	Hide	Created_by_Object_Filler_v2.2		
18	Net_10.10.11.0	net	10.10.11.0	255.255.255.0	blue	192.168.100.102	ngxr60	Hide	Created_by_Object_Filler_v2.2		
19	Net_10.10.12.0	net	10.10.12.0	255.255.255.0	blue	192.168.100.103	ngxr60	Hide	Created_by_Object_Filler_v2.2		
20	Net_10.10.13.0	net	10.10.13.0	255.255.255.0	blue	192.168.100.104	ngxr60	Hide	Created_by_Object_Filler_v2.2		
21	Net_10.10.14.0	net	10.10.14.0	255.255.255.0	blue	192.168.100.105	ngxr60	Hide	Created_by_Object_Filler_v2.2		
22	Net_10.10.15.0	net	10.10.15.0	255.255.255.0	blue	192.168.100.106	ngxr60	Hide	Created_by_Object_Filler_v2.2		
23	Net_10.10.16.0	net	10.10.16.0	255.255.255.0	blue	192.168.100.107	ngxr60	Hide	Created_by_Object_Filler_v2.2		
24	Net_10.10.17.0	net	10.10.17.0	255.255.255.0	blue	192.168.100.108	ngxr60	Hide	Created_by_Object_Filler_v2.2		
25	Net_10.10.18.0	net	10.10.18.0	255.255.255.0	blue	192.168.100.109	ngxr60	Hide	Created_by_Object_Filler_v2.2		
26	Net_10.10.19.0	net	10.10.19.0	255.255.255.0	blue	192.168.100.110	ngxr60	Hide	Created_by_Object_Filler_v2.2		
27	Net_10.10.20.0	net	10.10.20.0	255.255.255.0	blue	192.168.100.101	ngxr60	Hide	Created_by_Object_Filler_v2.2		
28	Net_10.10.21.0	net	10.10.21.0	255.255.255.0	blue	192.168.100.102	ngxr60	Hide	Created_by_Object_Filler_v2.2		

Your actual output may be slightly different, as I already had defined an InterSpect device in my case, but I’m leaving the output as is, so you can see how it looks.

Notice that when it comes to services, Object Dumper only exports the services you created, and not all the default (predefined) ones. That’s an interesting feature. If you want to have all the default objects exported as well, you may use the “`-d`” (*default*) switch in the Object Dumper’s command line, this way: **`odumper -f objects_5_0.C -o myobjects.csv -d`**

The nice thing about this file is that it can be used later as source for Object Filler so, as an example, all these objects can be carried to another SmartCenter or CMA and import them there. We will see later that and other cool things you can do with this information you have now…

By the way, be careful! These objects are part of the configuration on your security policy. If I were you, I’d watch very well where I leave them!.

6.2 Exporting network and service objects to a HTML File

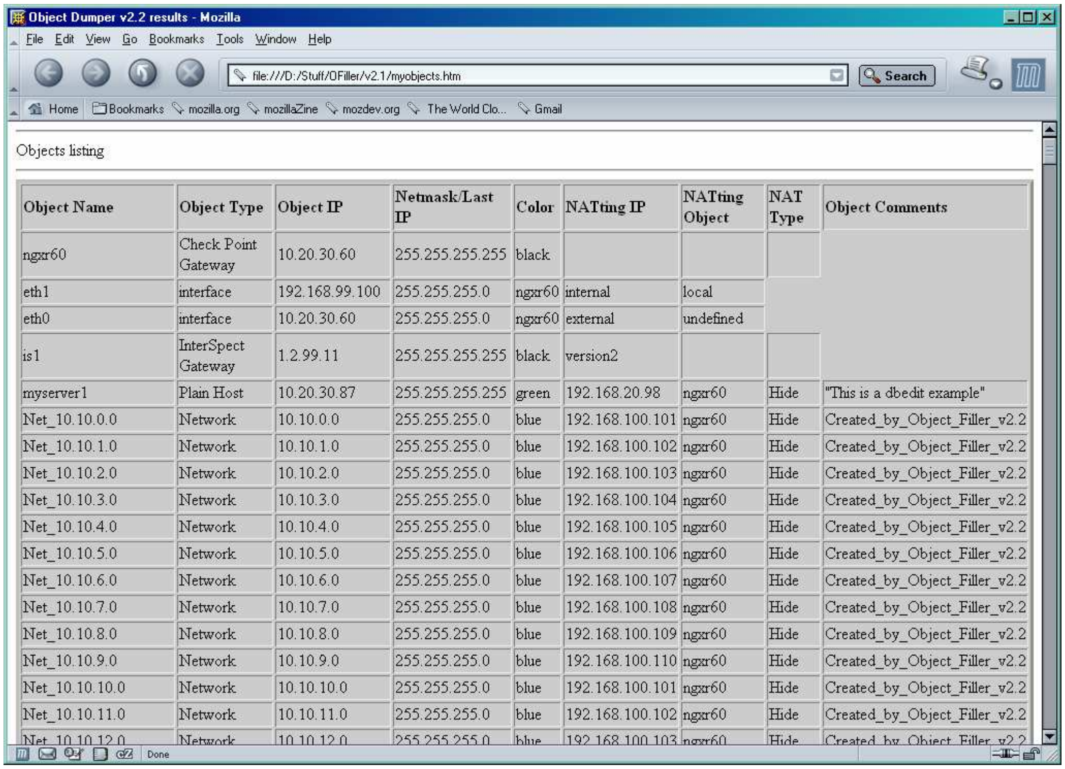
Before we finish this section, let’s say you enjoyed the report a lot, but now want to have it in HTML format. It is possible!

We assume you will use the same files you already have and used in the previous exercise. All you have to do then is to tell Object Dumper you prefer HTML (using the *–html* switch) and that’s it.

Let's see:

```
D:\Stuff\OFiller\v2.1>odumper -f objects_5_0.C -o myobjects.htm -html
Unofficial/Unsupported Object Dumper V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
==============================================================================
Processing objects...
................................................................................
................................................................................
....................................................
==============================================================================
Processed 42120 possible objects and found 500 valid ones.
From these, 264 were NATted records. It took 3 seconds on quiet mode.
Total successfully processed CP Gateways = 1
Total successfully processed Hosts = 172
Total successfully processed InterSpect Devices = 1
Total successfully processed Networks = 266
Total successfully processed Interfaces = 2
Total successfully processed TCP Services = 6
Total successfully processed UDP Services = 6
Task done successfully! - Thank you for using Object Dumper V2.2!
```

The result is more or less the same, with the exception that now the format is (a very ulgy) HTML page:


Object Dumper v2.2 results - Mozilla
File Edit View Go Bookmarks Tools Window Help
file:///D:/Stuff/OFiller/v2.1/myobjects.htm
Search
Home
Bookmarks
mozilla.org
mozillaZine
mozdev.org
The World Clo...
Gmail
Done

Objects listing

Object Name	Object Type	Object IP	Netmask/Last IP	Color	NATting IP	NATting Object	NAT Type	Object Comments
ngxr60	Check Point Gateway	10.20.30.60	255.255.255.255	black				
eth1	interface	192.168.99.100	255.255.255.0	ngxr60	internal	local		
eth0	interface	10.20.30.60	255.255.255.0	ngxr60	external	undefined		
is1	InterSpect Gateway	1.2.99.11	255.255.255.255	black	version2			
myserver1	Plain Host	10.20.30.87	255.255.255.255	green	192.168.20.98	ngxr60	Hide	"This is a dbedit example"
Net_10.10.0.0	Network	10.10.0.0	255.255.255.0	blue	192.168.100.101	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.1.0	Network	10.10.1.0	255.255.255.0	blue	192.168.100.102	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.2.0	Network	10.10.2.0	255.255.255.0	blue	192.168.100.103	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.3.0	Network	10.10.3.0	255.255.255.0	blue	192.168.100.104	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.4.0	Network	10.10.4.0	255.255.255.0	blue	192.168.100.105	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.5.0	Network	10.10.5.0	255.255.255.0	blue	192.168.100.106	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.6.0	Network	10.10.6.0	255.255.255.0	blue	192.168.100.107	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.7.0	Network	10.10.7.0	255.255.255.0	blue	192.168.100.108	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.8.0	Network	10.10.8.0	255.255.255.0	blue	192.168.100.109	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.9.0	Network	10.10.9.0	255.255.255.0	blue	192.168.100.110	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.10.0	Network	10.10.10.0	255.255.255.0	blue	192.168.100.101	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.11.0	Network	10.10.11.0	255.255.255.0	blue	192.168.100.102	ngxr60	Hide	Created_by_Object_Filler_v2.2
Net_10.10.12.0	Network	10.10.12.0	255.255.255.0	blue	192.168.100.103	ngxr60	Hide	Created_by_Object_Filler_v2.2

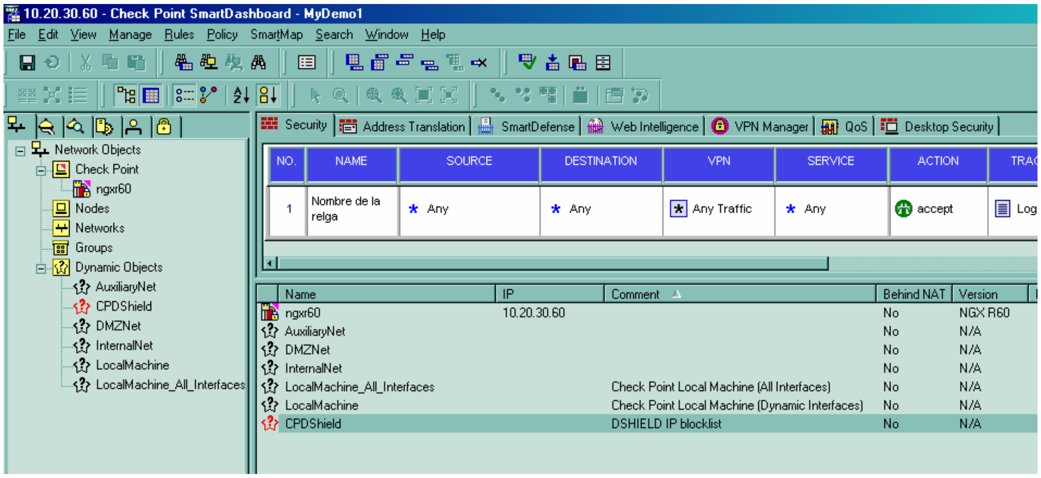
Cool!

7. Special (strange or non-common) operations with objects

7.1 Exporting from one place (SmartCenter or CMA) and importing back in the same or another place

WARNING: The next section assumes that you have the Object Dumper output of your configuration. If you don't, then right now either do a backup of your configuration (the way you know how), or get the Object Dumper output for the current configuration you have in your SmartCenter, following the steps described in section 6.1 – If you don't have this, then don't say later that I didn't warn you…

Okay. Let's do a crazy thing. Let's erase all the objects we created in all the previous steps. I'm not kidding. I mean it. Your SmartDashboard should look like this:



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRAC
1	Nombre de la relga	* Any	* Any	Any Traffic	* Any	accept	Log

Name	IP	Comment	Behind NAT	Version
ngxr60	10.20.30.60		No	NGX R60
AuxiliaryNet			No	N/A
DMZNet			No	N/A
InternalNet			No	N/A
LocalMachine_All_Interfaces		Check Point Local Machine (All Interfaces)	No	N/A
LocalMachine		Check Point Local Machine (Dynamic Interfaces)	No	N/A
CPDShield		DSHIELD IP blocklist	No	N/A

Well, or something similar to it.

Remember that you created some services also when you imported the Cisco PIX objects. Please delete them as well.

Now Edit the CSV File you got as output from Object Dumper. There, delete the objects that represent interfaces, the Check Point gateway object and any other object we didn't create as part of the previous exercises. We are doing this because those objects are still there, and we don't want to mess up with them.

IMPORTANT: Object Filler has no way to know if an object exists or not already in the SmartCenter. It is suggested, if you are importing these files in a SmartCenter that already contains objects, to make sure there are no duplicates by first exporting the objects lists with Object Dumper and then running Object Filler exclusively over the objects that are not present.

Once you did that, run Object Filler over the CSV file you have edited. Something like this:

```
D:\Stuff\OFiller\V2.2>ofiller -f myobjects_edited.csv -i csv -o myobjsagain.dbed
it
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
Processing objects...
................................................................................
..........
===============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 1152 possible objects and/or rules.
Found 450 total valid (or successfully processed) objects/rules.
Found 264 total NATted records.
-------------------------------------------------------------------------------
Total successfully processed Hosts = 172
Total successfully processed Networks = 266
Total successfully processed TCP Service objects = 6
Total successfully processed UDP Service objects = 6
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

Notice that the import type is now *csv*, so the *–i* swich is used as `–i csv`

Now you have a DBedit script containing all the commands you need to create the objects you deleted again. Please note that while we're importing back into the same SmartCenter, we could be doing this "restore" into a different SmartCenter

Yes, it is possible to do this restore even to a CMA in a Provider-1 environment. Yes, if you want to import those in the Global Objects Database of a Provider-1 MDS, is possible. There's documentation on how to do this in the same site where you can find this document posted (the ones mentioned at the very beginning). If you don't know what I just said in this paragraph, then never mind. ☺

Let's do it once again. First, we pass the file (in ASCII) to the SmartCenter

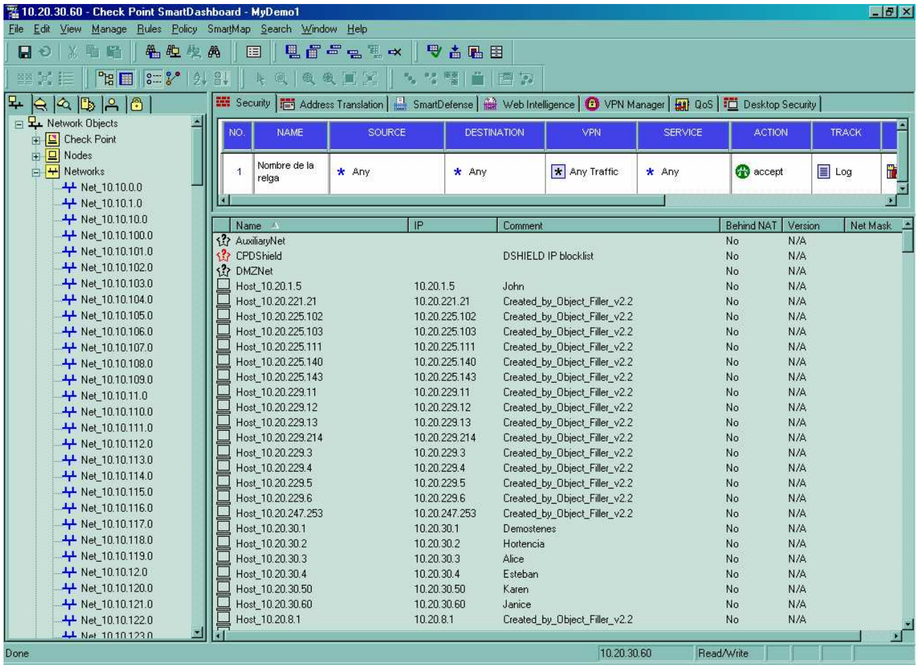
```
[Expert@ngxr60]# pwd
/home/admin
[Expert@ngxr60]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
```

```
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> get myobjsagain.dbedit
local: myobjsagain.dbedit remote: myobjsagain.dbedit
227 Entering passive mode (10,20,30,76,4,106)
125 Using existing data connection
################################################################################
##############################################################
226 Closing data connection; File transfer successful.
226642 bytes received in 0.23 secs (9.6e+02 Kbytes/sec)
ftp> bye
221 Service closing control connection
```

Then we run dbedit over the file we just transferred. Don't forget to check the objects database is not locked.

```
[Expert@ngxr60]# dbedit -f myobjsagain.dbedit -u admin -s localhost
Enter Administrator Password: secret
myserver1 updated successfully.
Net_10.10.0.0 updated successfully.
Net_10.10.1.0 updated successfully.
    .
    .
    .
UDP_Port_445 updated successfully.
UDP_Port_8998 updated successfully.
```

Wow! That was fast! Wasn't it? - Let's take a look on the SmartDashboard now…


10.20.30.60 - Check Point SmartDashboard - MyDemo1
File Edit View Manage Rules Policy SmartMap Search Window Help
Security
Address Translation
SmartDefense
Web Intelligence
VPN Manager
QoS
Desktop Security
Done
10.20.30.60
Read/Write

Network Objects
Check Point
Nodes
Networks
Net_10.10.0.0
Net_10.10.1.0
Net_10.10.10.0
Net_10.10.100.0
Net_10.10.101.0
Net_10.10.102.0
Net_10.10.103.0
Net_10.10.104.0
Net_10.10.105.0
Net_10.10.106.0
Net_10.10.107.0
Net_10.10.108.0
Net_10.10.109.0
Net_10.10.11.0
Net_10.10.110.0
Net_10.10.111.0
Net_10.10.112.0
Net_10.10.113.0
Net_10.10.114.0
Net_10.10.115.0
Net_10.10.116.0
Net_10.10.117.0
Net_10.10.118.0
Net_10.10.119.0
Net_10.10.12.0
Net_10.10.120.0
Net_10.10.121.0
Net_10.10.122.0
Net_10.10.123.0

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK
1	Nombre de la relga	* Any	* Any	* Any Traffic	* Any	accept	Log

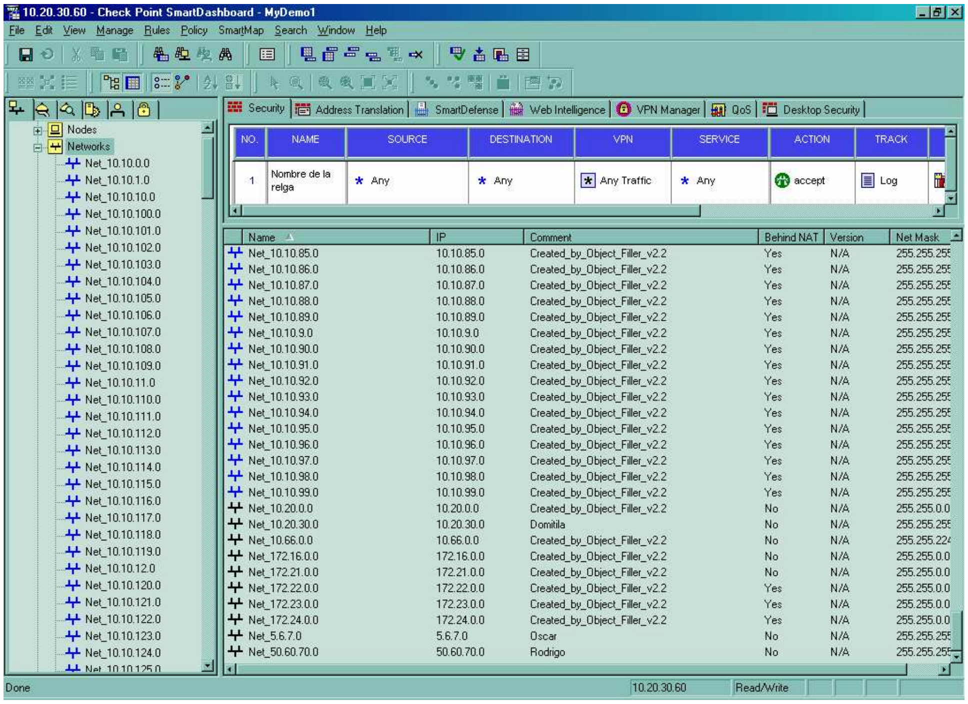
Name	IP	Comment	Behind NAT	Version	Net Mask
AuxiliaryNet			No	N/A	
CPDShield		DSHIELD IP blocklist	No	N/A	
DMZNet			No	N/A	
Host_10.20.1.5	10.20.1.5	John	No	N/A	
Host_10.20.221.21	10.20.221.21	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.225.102	10.20.225.102	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.225.103	10.20.225.103	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.225.111	10.20.225.111	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.225.140	10.20.225.140	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.225.143	10.20.225.143	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.11	10.20.229.11	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.12	10.20.229.12	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.13	10.20.229.13	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.214	10.20.229.214	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.3	10.20.229.3	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.4	10.20.229.4	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.5	10.20.229.5	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.229.6	10.20.229.6	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.247.253	10.20.247.253	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.20.30.1	10.20.30.1	Demostenes	No	N/A	
Host_10.20.30.2	10.20.30.2	Hortencia	No	N/A	
Host_10.20.30.3	10.20.30.3	Alice	No	N/A	
Host_10.20.30.4	10.20.30.4	Esteban	No	N/A	
Host_10.20.30.50	10.20.30.50	Karen	No	N/A	
Host_10.20.30.60	10.20.30.60	Janice	No	N/A	
Host_10.20.8.1	10.20.8.1	Created_by_Object_Filler_v2.2	No	N/A	

Yes, it seems that all is there again… Cool!

7.2 Changing name to multiple objects

Now let’s do some magic. Let’s say that you want to rename all the *Net_** objects and you want them to be named *Network_** - that’s a better name after all, isn’t it?

This is the way SmartDashboard originally looks like



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK
1	Nombre de la relga	* Any	* Any	Any Traffic	* Any	accept	Log

Name	IP	Comment	Behind NAT	Version	Net Mask
Net_10.10.85.0	10.10.85.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.86.0	10.10.86.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.87.0	10.10.87.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.88.0	10.10.88.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.89.0	10.10.89.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.9.0	10.10.9.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.90.0	10.10.90.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.91.0	10.10.91.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.92.0	10.10.92.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.93.0	10.10.93.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.94.0	10.10.94.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.95.0	10.10.95.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.96.0	10.10.96.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.97.0	10.10.97.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.98.0	10.10.98.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.10.99.0	10.10.99.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Net_10.20.0.0	10.20.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Net_10.20.30.0	10.20.30.0	Domitila	No	N/A	255.255.255
Net_10.66.0.0	10.66.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.224
Net_172.16.0.0	172.16.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Net_172.21.0.0	172.21.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Net_172.22.0.0	172.22.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Net_172.23.0.0	172.23.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Net_172.24.0.0	172.24.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Net_5.6.7.0	5.6.7.0	Oscar	No	N/A	255.255.255
Net_50.60.70.0	50.60.70.0	Rodrigo	No	N/A	255.255.255

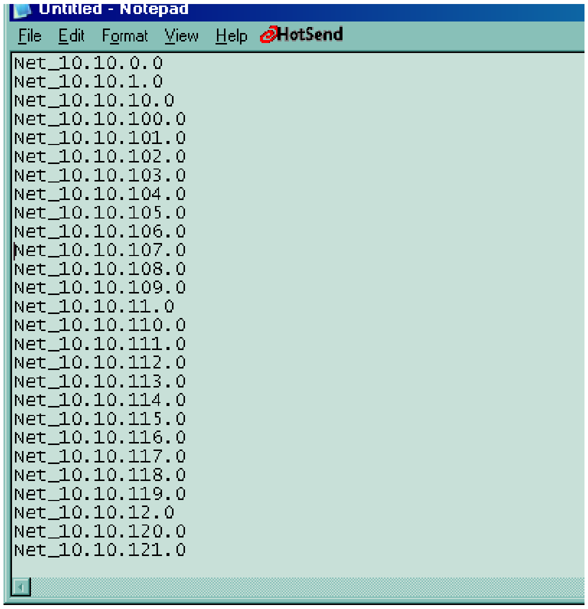
Now, let’s begin! We’ll use for this the CSV file that we got when we exported the objects with Object Dumper. Order all the items by the first column (Which is name) and then delete all the objects that are not networks.

Then delete all the columns you have there. Just leave the first column (*Column A*) in such spreadsheet.

Now, copy the whole first column to a text editor. Make sure all the rows were copied.

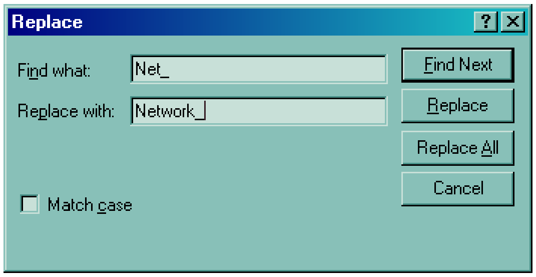
Notepad can do this job, but if you prefer something else you can do it with something else. I’ll use Notepad because is available in every Windows machine.

Your Notepad should look like this:



```
Net_10.10.0.0
Net_10.10.1.0
Net_10.10.10.0
Net_10.10.100.0
Net_10.10.101.0
Net_10.10.102.0
Net_10.10.103.0
Net_10.10.104.0
Net_10.10.105.0
Net_10.10.106.0
Net_10.10.107.0
Net_10.10.108.0
Net_10.10.109.0
Net_10.10.11.0
Net_10.10.110.0
Net_10.10.111.0
Net_10.10.112.0
Net_10.10.113.0
Net_10.10.114.0
Net_10.10.115.0
Net_10.10.116.0
Net_10.10.117.0
Net_10.10.118.0
Net_10.10.119.0
Net_10.10.12.0
Net_10.10.120.0
Net_10.10.121.0
```

Then, using the Search and Replace functionality of your text editor, change all the "*Net_**" occurrences by "*Network_**". The Search and Replace window should look like this:



Find what: Net_

Replace with: Network_

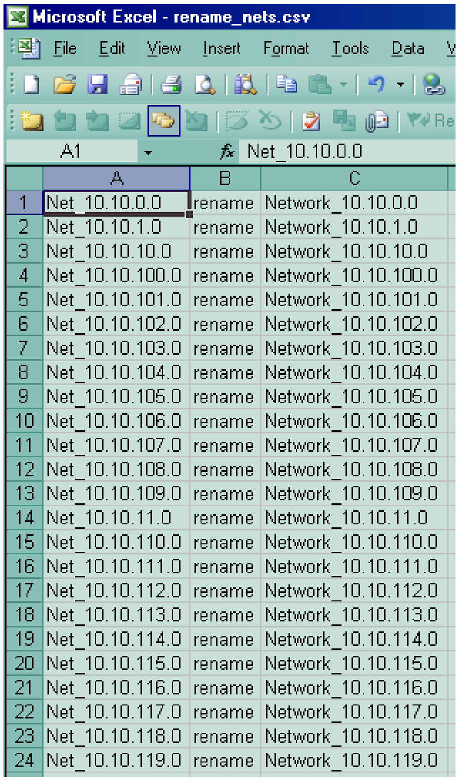
□ Match case

The operation performed should leave all your names changed the way you want them to be.

Now, put all those names in the **third** column of your spreadsheet.

Finally, fill out the second columns with the word `rename`. You need to type it just once, and then just copy-paste it all over the place.

The resulting spreadsheet should look like the one below. Save it as CSV file.


Microsoft Excel - rename_nets.csv
File Edit View Insert Format Tools Data
A1
Net_10.10.0.0

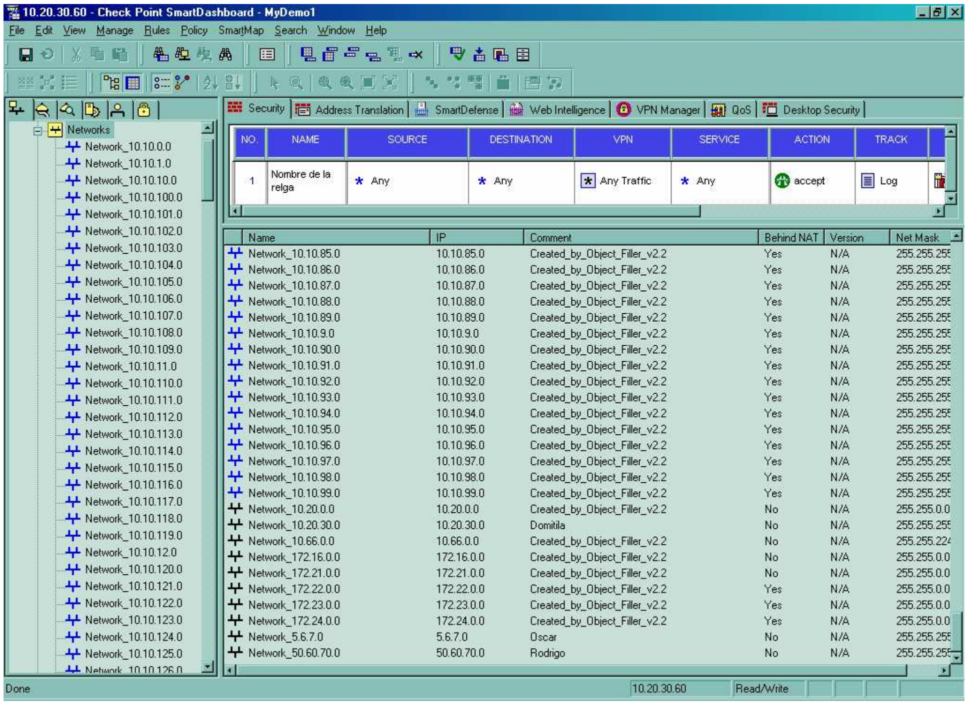
	A	B	C
1	Net_10.10.0.0	rename	Network_10.10.0.0
2	Net_10.10.1.0	rename	Network_10.10.1.0
3	Net_10.10.10.0	rename	Network_10.10.10.0
4	Net_10.10.100.0	rename	Network_10.10.100.0
5	Net_10.10.101.0	rename	Network_10.10.101.0
6	Net_10.10.102.0	rename	Network_10.10.102.0
7	Net_10.10.103.0	rename	Network_10.10.103.0
8	Net_10.10.104.0	rename	Network_10.10.104.0
9	Net_10.10.105.0	rename	Network_10.10.105.0
10	Net_10.10.106.0	rename	Network_10.10.106.0
11	Net_10.10.107.0	rename	Network_10.10.107.0
12	Net_10.10.108.0	rename	Network_10.10.108.0
13	Net_10.10.109.0	rename	Network_10.10.109.0
14	Net_10.10.11.0	rename	Network_10.10.11.0
15	Net_10.10.110.0	rename	Network_10.10.110.0
16	Net_10.10.111.0	rename	Network_10.10.111.0
17	Net_10.10.112.0	rename	Network_10.10.112.0
18	Net_10.10.113.0	rename	Network_10.10.113.0
19	Net_10.10.114.0	rename	Network_10.10.114.0
20	Net_10.10.115.0	rename	Network_10.10.115.0
21	Net_10.10.116.0	rename	Network_10.10.116.0
22	Net_10.10.117.0	rename	Network_10.10.117.0
23	Net_10.10.118.0	rename	Network_10.10.118.0
24	Net_10.10.119.0	rename	Network_10.10.119.0

Then run Object Filler over it using *csv* as import type (`-i csv`)

```
D:\Stuff\OFiller\V2.2>ofiller -f rename_nets.csv -i csv -o renames.dbedit
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
Processing objects...
.....................................................
===============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 266 possible objects and/or rules.
Found 266 total valid (or successfully processed) objects/rules.
-------------------------------------------------------------------------------
Total successfully processed RENAME statements = 266
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

Finally import the resulting file into SmartCenter. Notice that DBedit now answers with the message `The Operation Finished Successfully` instead of the regular `XXXXXX updated successfully`. This is because you are performing operations over the objects, and not modifying the object's properties themselves.

The SmartDashboard should now look like this:



Networks
- Network_10.10.0.0
- Network_10.10.1.0
- Network_10.10.10.0
- Network_10.10.100.0
- Network_10.10.101.0
- Network_10.10.102.0
- Network_10.10.103.0
- Network_10.10.104.0
- Network_10.10.105.0
- Network_10.10.106.0
- Network_10.10.107.0
- Network_10.10.108.0
- Network_10.10.109.0
- Network_10.10.11.0
- Network_10.10.110.0
- Network_10.10.111.0
- Network_10.10.112.0
- Network_10.10.113.0
- Network_10.10.114.0
- Network_10.10.115.0
- Network_10.10.116.0
- Network_10.10.117.0
- Network_10.10.118.0
- Network_10.10.119.0
- Network_10.10.12.0
- Network_10.10.120.0
- Network_10.10.121.0
- Network_10.10.122.0
- Network_10.10.123.0
- Network_10.10.124.0
- Network_10.10.125.0
- Network_10.10.126.0

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK
1	Nombre de la relga	* Any	* Any	* Any Traffic	* Any	accept	Log

Name	IP	Comment	Behind NAT	Version	Net Mask
Network_10.10.85.0	10.10.85.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.86.0	10.10.86.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.87.0	10.10.87.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.88.0	10.10.88.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.89.0	10.10.89.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.9.0	10.10.9.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.90.0	10.10.90.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.91.0	10.10.91.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.92.0	10.10.92.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.93.0	10.10.93.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.94.0	10.10.94.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.95.0	10.10.95.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.96.0	10.10.96.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.97.0	10.10.97.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.98.0	10.10.98.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.10.99.0	10.10.99.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.255
Network_10.20.0.0	10.20.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Network_10.20.30.0	10.20.30.0	Domitila	No	N/A	255.255.255
Network_10.66.0.0	10.66.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.224
Network_172.16.0.0	172.16.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Network_172.21.0.0	172.21.0.0	Created_by_Object_Filler_v2.2	No	N/A	255.255.0.0
Network_172.22.0.0	172.22.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Network_172.23.0.0	172.23.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Network_172.24.0.0	172.24.0.0	Created_by_Object_Filler_v2.2	Yes	N/A	255.255.0.0
Network_5.6.7.0	5.6.7.0	Oscar	No	N/A	255.255.255
Network_50.60.70.0	50.60.70.0	Rodrigo	No	N/A	255.255.255

Magic!

7.3 Changing IP address to multiple objects

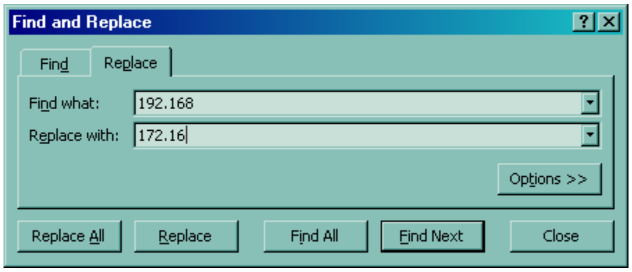
Now let’s assume you want to change all the hosts that belong to the 192.168.220.0/24 subnet, to the 172.16.220.0/24 subnet, because you are reorganizing the whole IP addressing in your WAN.

Piece of cake.

First open again the CSV file that resulted of running Object Dumper. Remote all the rows that don’t belong to the objects you want to modify. Just leave the host objects you are interested on changing.

Then, run “Search and Replace” in Excel. Search for “192.168” and replace with “172.16”. But… Careful! – Do it **only** over the IP address column. If you do it in the whole document, you will change the name of the objects as well, and then the magic won’t result, as we will be referencing to a non-existent object.

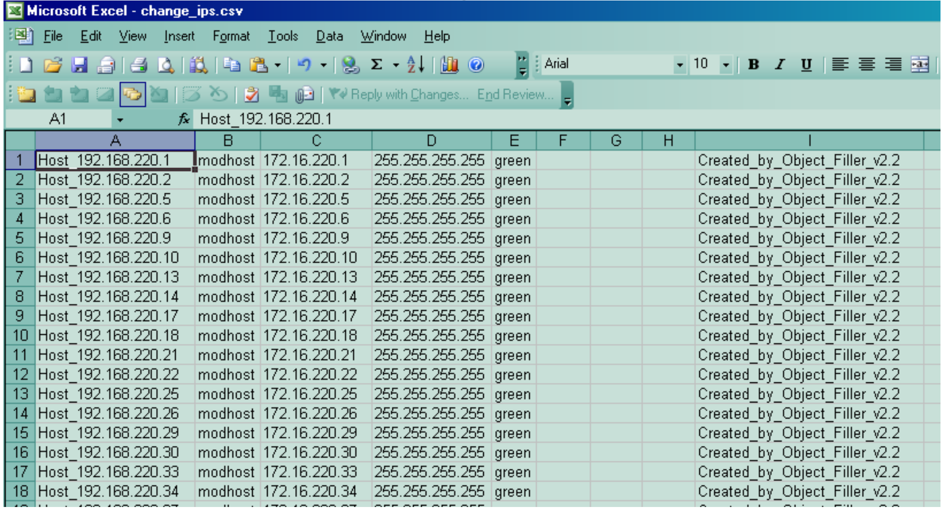
The Search and Replace window should look like this



Find what: 192.168
Replace with: 172.16

Now, in the second column, change the word “host” by the word “modhost”, which instead of creating a new object, just modifies it…

The final result should be something like this:



A1 Host_192.168.220.1

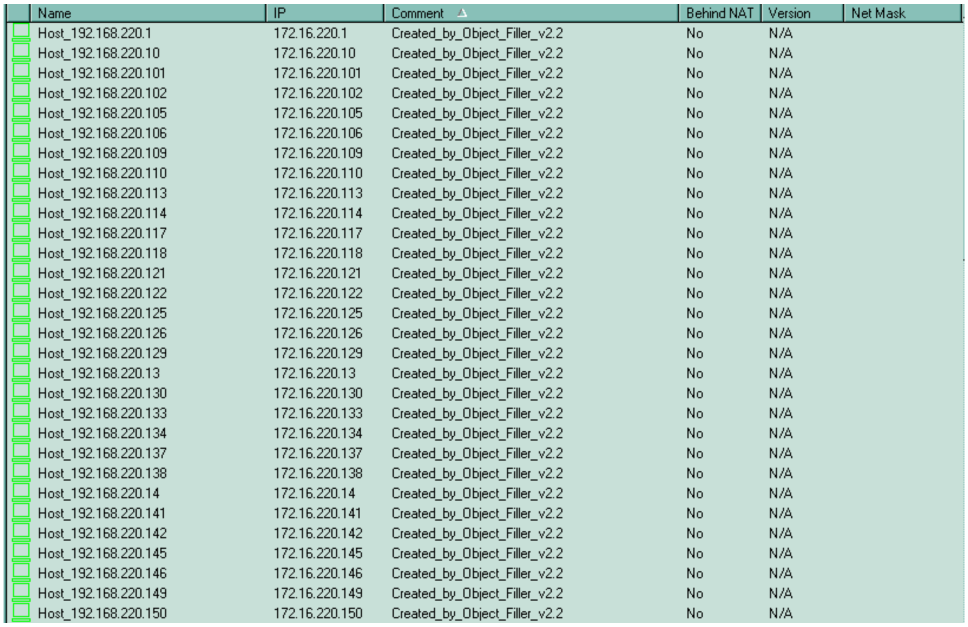
	A	B	C	D	E	F	G	H	I
1	Host_192.168.220.1	modhost	172.16.220.1	255.255.255.255	green				Created_by_Object_Filler_v2.2
2	Host_192.168.220.2	modhost	172.16.220.2	255.255.255.255	green				Created_by_Object_Filler_v2.2
3	Host_192.168.220.5	modhost	172.16.220.5	255.255.255.255	green				Created_by_Object_Filler_v2.2
4	Host_192.168.220.6	modhost	172.16.220.6	255.255.255.255	green				Created_by_Object_Filler_v2.2
5	Host_192.168.220.9	modhost	172.16.220.9	255.255.255.255	green				Created_by_Object_Filler_v2.2
6	Host_192.168.220.10	modhost	172.16.220.10	255.255.255.255	green				Created_by_Object_Filler_v2.2
7	Host_192.168.220.13	modhost	172.16.220.13	255.255.255.255	green				Created_by_Object_Filler_v2.2
8	Host_192.168.220.14	modhost	172.16.220.14	255.255.255.255	green				Created_by_Object_Filler_v2.2
9	Host_192.168.220.17	modhost	172.16.220.17	255.255.255.255	green				Created_by_Object_Filler_v2.2
10	Host_192.168.220.18	modhost	172.16.220.18	255.255.255.255	green				Created_by_Object_Filler_v2.2
11	Host_192.168.220.21	modhost	172.16.220.21	255.255.255.255	green				Created_by_Object_Filler_v2.2
12	Host_192.168.220.22	modhost	172.16.220.22	255.255.255.255	green				Created_by_Object_Filler_v2.2
13	Host_192.168.220.25	modhost	172.16.220.25	255.255.255.255	green				Created_by_Object_Filler_v2.2
14	Host_192.168.220.26	modhost	172.16.220.26	255.255.255.255	green				Created_by_Object_Filler_v2.2
15	Host_192.168.220.29	modhost	172.16.220.29	255.255.255.255	green				Created_by_Object_Filler_v2.2
16	Host_192.168.220.30	modhost	172.16.220.30	255.255.255.255	green				Created_by_Object_Filler_v2.2
17	Host_192.168.220.33	modhost	172.16.220.33	255.255.255.255	green				Created_by_Object_Filler_v2.2
18	Host_192.168.220.34	modhost	172.16.220.34	255.255.255.255	green				Created_by_Object_Filler_v2.2

Save this spreadsheet with CSV format. Run Object Filler over it using `-i csv` as import type.

```
D:\Stuff\OFiller\V2.2>ofiller -f change_ips.csv -i csv -o change_ips.dbedit
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
==============================================================================
Processing objects...
.........................
==============================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 256 possible objects and/or rules.
Found 128 total valid (or successfully processed) objects/rules.
------------------------------------------------------------------------------
Total successfully processed Hosts = 128
------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
==============================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

And then import the result into SmartCenter using DBedit the way we did it before.

The result appears now in the SmartDashboard:



Name	IP	Comment	Behind NAT	Version	Net Mask
Host_192.168.220.1	172.16.220.1	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.10	172.16.220.10	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.101	172.16.220.101	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.102	172.16.220.102	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.105	172.16.220.105	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.106	172.16.220.106	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.109	172.16.220.109	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.110	172.16.220.110	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.113	172.16.220.113	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.114	172.16.220.114	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.117	172.16.220.117	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.118	172.16.220.118	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.121	172.16.220.121	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.122	172.16.220.122	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.125	172.16.220.125	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.126	172.16.220.126	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.129	172.16.220.129	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.13	172.16.220.13	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.130	172.16.220.130	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.133	172.16.220.133	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.134	172.16.220.134	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.137	172.16.220.137	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.138	172.16.220.138	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.14	172.16.220.14	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.141	172.16.220.141	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.142	172.16.220.142	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.145	172.16.220.145	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.146	172.16.220.146	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.149	172.16.220.149	Created_by_Object_Filler_v2.2	No	N/A	
Host_192.168.220.150	172.16.220.150	Created_by_Object_Filler_v2.2	No	N/A	

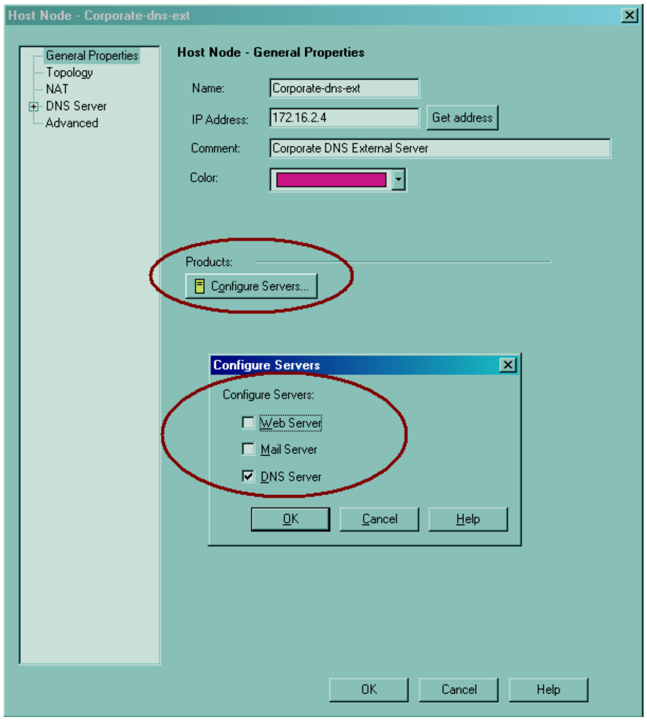
If you would like to change the name as well, you already know how to do so… ;-)

7.4 Adding Web, Mail and/or DNS Server properties to Host Objects (Node, Host) with NG+AI R55, R55W and NGX R60-R62

In Check Point Next Generation with Application Intelligence R55 (NG+AI R55 for short). We had the chance of specifying if a Host object was actually a Web Server. Enabling the Web Server property allowed the firewall to apply certain very specific protections against Web Attacks. In the following release (NG+AI R55W) the Web Intelligence set of protections was born. As a side comment, since NG+AI R55W some of the Web Intelligence protections require a license. Please consult your Product Documentation to find more about it.

Since NG+AI R55W we are as well able specify if a Host object is a DNS or Mail Server. Setting such properties gives very granular protections against application attacks that go against those services. Those protections don’t require a license so far.

Below you can see a graphic of the Hosts Properties where you may define the Web, Mail or DNS Server attributes for a single object within SmartDashboard.



Host Node - General Properties

Name: Corporate-dns-ext
IP Address: 172.16.2.4
Comment: Corporate DNS External Server
Color: 🟥

Configure Servers

Configure Servers:
□ Web Server
□ Mail Server
☑ DNS Server

So, there are very good things enabling the Web, Mail and or DNS Servers property to the Host objects – actually you should, or you'd be missing very, very, very good protection to your infrastructure on not doing so… - and if you haven't read about all this, go and take a look at the manuals right now. I assure you it won't be a waste of time at all!

The catch: You have to do it one-by-one. If you have just 10 Hosts objects for which you need to enable this, is not even worth for you continuing reading further this section. You probably will be done doing it manually by the time you finish reading this text! – A different story is if you have more than 50 (perhaps 100 or 200) servers for which you need to enable these settings… Doing that one by one would be really painful for your fingers and would require lots of time. Once again, that is the case when Object Filler comes in handy.

Before we go ahead, you have to know that we'll do the testing using NGX R60A because that's what I have installed at the time I'm writing this. It works without problems with NG+AI R55W and NGX R60. DNS and Mail Server properties DON'T work before NG+AI R55W. Web Server property DOESN'T work before NG+AI R55 – you are warned now…

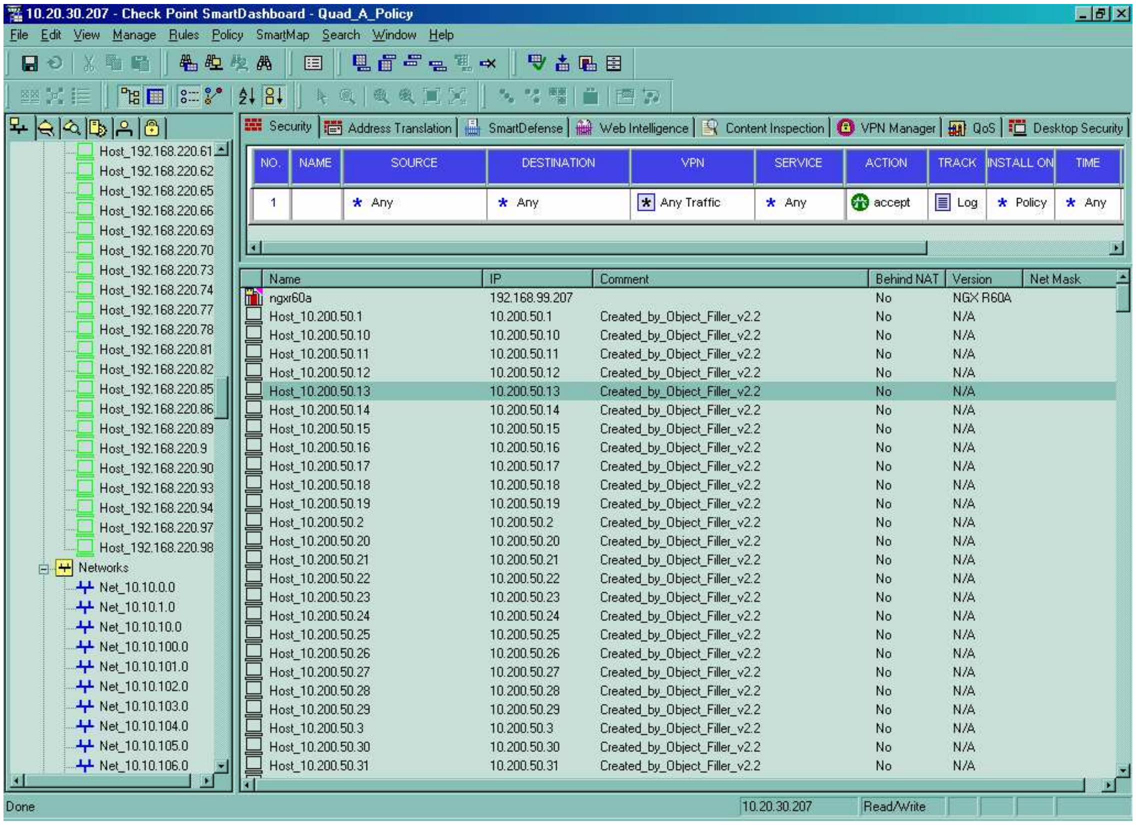
To begin with, let's assume you already have some hosts created in your SmartCenter. Let's assume those objects are within the 10.200.50.0/24 netmask range going from 10.200.50.1 to 10.200.50.50 - Since probably you don't have them already created, you may use the following Object Filler sentence to do so:

```
D:\Stuff\OFiller\V2.2>ofiller -s 10.200.50.1 -d 10.200.50.50 -m 24 -t hosts –o hosts_for_srv.dbedit
```

You should know by now how to import them. ;-)

Now, let’s assume as well you still have in your SmartCenter the objects created in the sections 5.1.1 and 5.1.2 – this is a bunch of Host and Network objects that have nothing to do with the exercise we will do, but that help to understand how to treat objects that have nothing to do with what we are doing, as for sure that would be the case in real life.

Your SmartDashboard may look more or less like this:

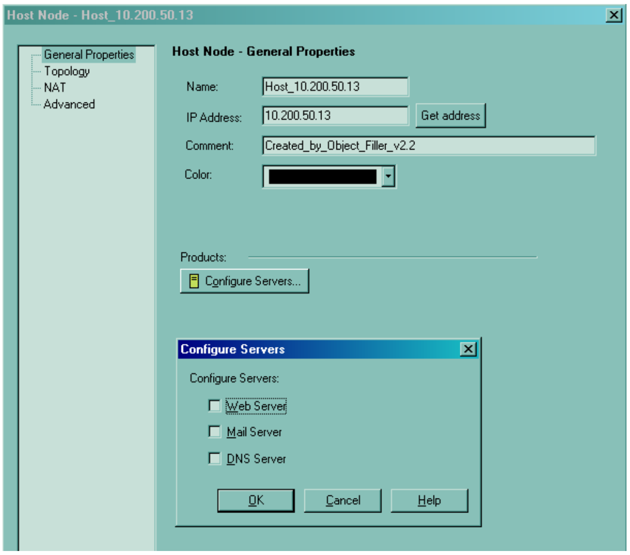


Host_192.168.220.61
Host_192.168.220.62
Host_192.168.220.65
Host_192.168.220.66
Host_192.168.220.69
Host_192.168.220.70
Host_192.168.220.73
Host_192.168.220.74
Host_192.168.220.77
Host_192.168.220.78
Host_192.168.220.81
Host_192.168.220.82
Host_192.168.220.85
Host_192.168.220.86
Host_192.168.220.89
Host_192.168.220.9
Host_192.168.220.90
Host_192.168.220.93
Host_192.168.220.94
Host_192.168.220.97
Host_192.168.220.98
Networks
Net_10.10.0.0
Net_10.10.1.0
Net_10.10.10.0
Net_10.10.100.0
Net_10.10.101.0
Net_10.10.102.0
Net_10.10.103.0
Net_10.10.104.0
Net_10.10.105.0
Net_10.10.106.0

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME
1		* Any	* Any	* Any Traffic	* Any	accept	Log	* Policy	* Any

Name	IP	Comment	Behind NAT	Version	Net Mask
ngxr60a	192.168.99.207		No	NGX R60A	
Host_10.200.50.1	10.200.50.1	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.10	10.200.50.10	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.11	10.200.50.11	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.12	10.200.50.12	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.13	10.200.50.13	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.14	10.200.50.14	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.15	10.200.50.15	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.16	10.200.50.16	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.17	10.200.50.17	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.18	10.200.50.18	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.19	10.200.50.19	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.2	10.200.50.2	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.20	10.200.50.20	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.21	10.200.50.21	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.22	10.200.50.22	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.23	10.200.50.23	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.24	10.200.50.24	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.25	10.200.50.25	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.26	10.200.50.26	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.27	10.200.50.27	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.28	10.200.50.28	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.29	10.200.50.29	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.3	10.200.50.3	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.30	10.200.50.30	Created_by_Object_Filler_v2.2	No	N/A	
Host_10.200.50.31	10.200.50.31	Created_by_Object_Filler_v2.2	No	N/A	

As we can see, so far none of them has the Mail, DNS or WebServer property set:



Name: Host_10.200.50.13

IP Address: 10.200.50.13

Comment: Created_by_Object_Filler_v2.2

□ Web Server

□ Mail Server

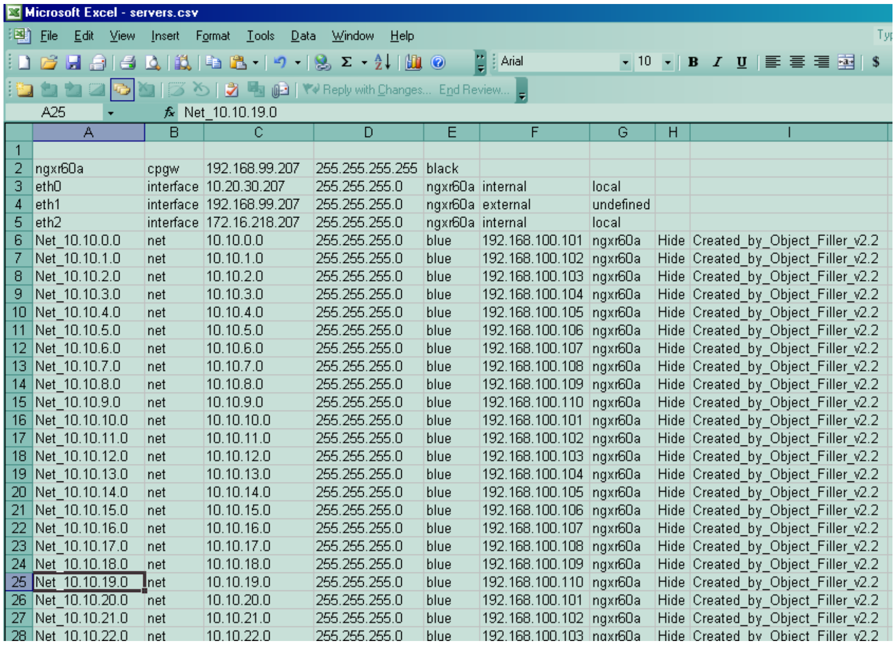
□ DNS Server

Now, we will use the following convention:
10.200.50.1 – 10.200.50.10 will be marked as DNS Servers
10.200.50.11 – 10.200.50.20 will be marked as Mail Servers
10.200.50.21 – 10.200.50.30 will be marked as Web Servers
10.200.50.31 – 10.200.50.40 will be marked as DNS and Mail Servers
10.200.50.41 – 10.200.50.50 will be marked as DNS, Mail and Web Servers

So, the first step we do here is to dump the Objects using Object Dumper. We need to pass the file to the Windows machine (as we did in the previous exercises) and then proceed. For convenience, we renamed the *objects_5_0.C* file to *objects_5_0_servers.C*, so we can differentiate it.

```
D:\Stuff\OFiller\v2.1>odumper -f objects_5_0_servers.C -o servers.csv
Unofficial/Unsupported Object Dumper V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===============================================================================
Processing objects...
................................................................................
................................................................................
...................................
===============================================================================
Processed 42659 possible objects and found 483 valid ones.
From these, 256 were NATted records. It took 3 seconds on quiet mode.
Total successfully processed CP Gateways = 1
Total successfully processed Hosts = 178
Total successfully processed Networks = 256
Total successfully processed Interfaces = 3
Task done successfully! - Thank you for using Object Dumper V2.2!
```

From this, we get a listing that begins more or less like the following screenshot



Microsoft Excel - servers.csv
File Edit View Insert Format Tools Data Window Help
Arial
10
Reply with Changes... End Review...
A25

Net_10.10.19.0

	A	B	C	D	E	F	G	H	I
1									
2	ngxr60a	cpgw	192.168.99.207	255.255.255.255	black				
3	eth0	interface	10.20.30.207	255.255.255.0	ngxr60a	internal	local		
4	eth1	interface	192.168.99.207	255.255.255.0	ngxr60a	external	undefined		
5	eth2	interface	172.16.218.207	255.255.255.0	ngxr60a	internal	local		
6	Net_10.10.0.0	net	10.10.0.0	255.255.255.0	blue	192.168.100.101	ngxr60a	Hide	Created_by_Object_Filler_v2.2
7	Net_10.10.1.0	net	10.10.1.0	255.255.255.0	blue	192.168.100.102	ngxr60a	Hide	Created_by_Object_Filler_v2.2
8	Net_10.10.2.0	net	10.10.2.0	255.255.255.0	blue	192.168.100.103	ngxr60a	Hide	Created_by_Object_Filler_v2.2
9	Net_10.10.3.0	net	10.10.3.0	255.255.255.0	blue	192.168.100.104	ngxr60a	Hide	Created_by_Object_Filler_v2.2
10	Net_10.10.4.0	net	10.10.4.0	255.255.255.0	blue	192.168.100.105	ngxr60a	Hide	Created_by_Object_Filler_v2.2
11	Net_10.10.5.0	net	10.10.5.0	255.255.255.0	blue	192.168.100.106	ngxr60a	Hide	Created_by_Object_Filler_v2.2
12	Net_10.10.6.0	net	10.10.6.0	255.255.255.0	blue	192.168.100.107	ngxr60a	Hide	Created_by_Object_Filler_v2.2
13	Net_10.10.7.0	net	10.10.7.0	255.255.255.0	blue	192.168.100.108	ngxr60a	Hide	Created_by_Object_Filler_v2.2
14	Net_10.10.8.0	net	10.10.8.0	255.255.255.0	blue	192.168.100.109	ngxr60a	Hide	Created_by_Object_Filler_v2.2
15	Net_10.10.9.0	net	10.10.9.0	255.255.255.0	blue	192.168.100.110	ngxr60a	Hide	Created_by_Object_Filler_v2.2
16	Net_10.10.10.0	net	10.10.10.0	255.255.255.0	blue	192.168.100.101	ngxr60a	Hide	Created_by_Object_Filler_v2.2
17	Net_10.10.11.0	net	10.10.11.0	255.255.255.0	blue	192.168.100.102	ngxr60a	Hide	Created_by_Object_Filler_v2.2
18	Net_10.10.12.0	net	10.10.12.0	255.255.255.0	blue	192.168.100.103	ngxr60a	Hide	Created_by_Object_Filler_v2.2
19	Net_10.10.13.0	net	10.10.13.0	255.255.255.0	blue	192.168.100.104	ngxr60a	Hide	Created_by_Object_Filler_v2.2
20	Net_10.10.14.0	net	10.10.14.0	255.255.255.0	blue	192.168.100.105	ngxr60a	Hide	Created_by_Object_Filler_v2.2
21	Net_10.10.15.0	net	10.10.15.0	255.255.255.0	blue	192.168.100.106	ngxr60a	Hide	Created_by_Object_Filler_v2.2
22	Net_10.10.16.0	net	10.10.16.0	255.255.255.0	blue	192.168.100.107	ngxr60a	Hide	Created_by_Object_Filler_v2.2
23	Net_10.10.17.0	net	10.10.17.0	255.255.255.0	blue	192.168.100.108	ngxr60a	Hide	Created_by_Object_Filler_v2.2
24	Net_10.10.18.0	net	10.10.18.0	255.255.255.0	blue	192.168.100.109	ngxr60a	Hide	Created_by_Object_Filler_v2.2
25	Net_10.10.19.0	net	10.10.19.0	255.255.255.0	blue	192.168.100.110	ngxr60a	Hide	Created_by_Object_Filler_v2.2
26	Net_10.10.20.0	net	10.10.20.0	255.255.255.0	blue	192.168.100.101	ngxr60a	Hide	Created_by_Object_Filler_v2.2
27	Net_10.10.21.0	net	10.10.21.0	255.255.255.0	blue	192.168.100.102	ngxr60a	Hide	Created_by_Object_Filler_v2.2
28	Net_10.10.22.0	net	10.10.22.0	255.255.255.0	blue	192.168.100.103	ngxr60a	Hide	Created_by_Object_Filler_v2.2

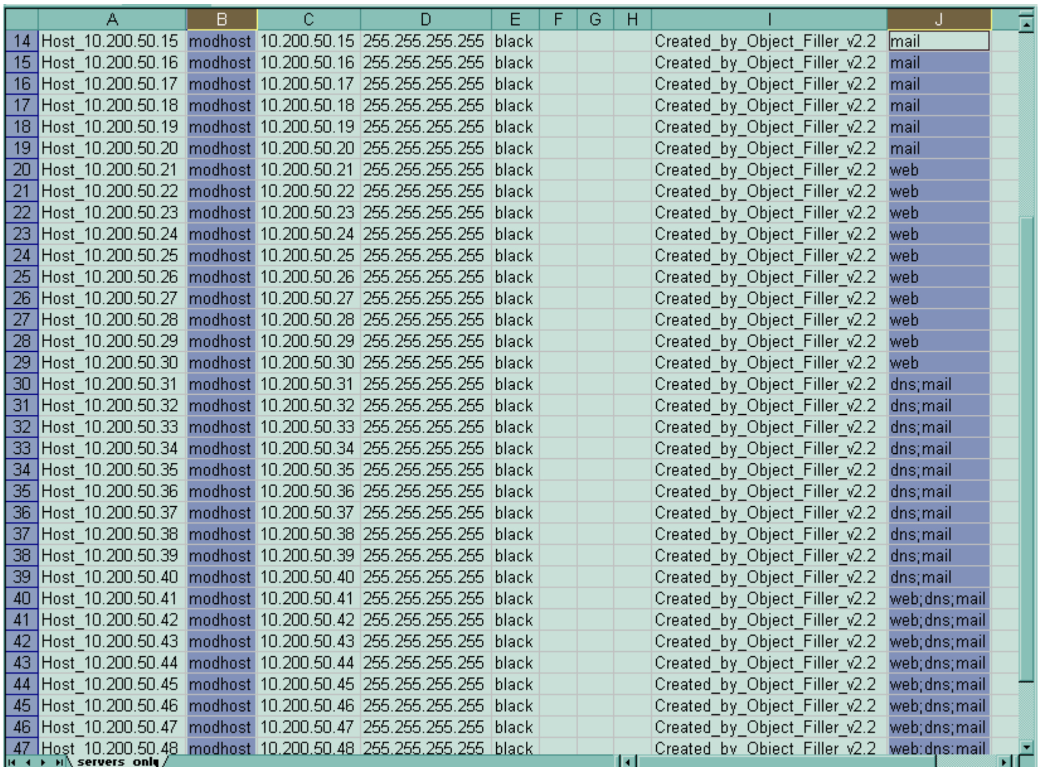
Well, let’s delete all the objects we won’t modify – so we need to leave the rows that include the Host Objects that represent the servers we are going to process.

Then change the column 2 (marked as B), substituting “*host*” with “*modhost*” – this is important so Object Filler knows that we are not creating the object, but actually just modifying some properties.

Then in the column 10 (marked as J) in the spreadsheet program, do the following:
For the objects from 10.200.50.1 to 10.200.50.10, write the word ***`dns`***
For the objects from 10.200.50.11 to 10.200.50.20, write the word ***`mail`***
For the objects from 10.200.50.21 to 10.200.50.30, write the word ***`web`***
For the objects from 10.200.50.31 to 10.200.50.40, write the words ***`dns;mail`*** (Note the semicolon separating both words, this is important)
For the objects from 10.200.50.41 to 10.200.50.50, write the words ***`web;dns;mail`*** (Note the semicolon separating the words, this is important)

It is important to know that the Web and DNS servers have as well a property to set what firewall is protecting them (i.e. behind what gateway such servers are placed). If you want to set those servers, you may use column 11 (column K) to set the gateway protecting the web servers and column 12 (column L) to set the gateway protecting the DNS server.

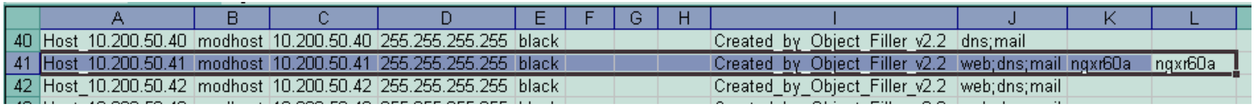
The initial result should look like the following screenshot:



	A	B	C	D	E	F	G	H	I	J
14	Host_10.200.50.15	modhost	10.200.50.15	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
15	Host_10.200.50.16	modhost	10.200.50.16	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
16	Host_10.200.50.17	modhost	10.200.50.17	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
17	Host_10.200.50.18	modhost	10.200.50.18	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
18	Host_10.200.50.19	modhost	10.200.50.19	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
19	Host_10.200.50.20	modhost	10.200.50.20	255.255.255.255	black				Created_by_Object_Filler_v2.2	mail
20	Host_10.200.50.21	modhost	10.200.50.21	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
21	Host_10.200.50.22	modhost	10.200.50.22	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
22	Host_10.200.50.23	modhost	10.200.50.23	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
23	Host_10.200.50.24	modhost	10.200.50.24	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
24	Host_10.200.50.25	modhost	10.200.50.25	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
25	Host_10.200.50.26	modhost	10.200.50.26	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
26	Host_10.200.50.27	modhost	10.200.50.27	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
27	Host_10.200.50.28	modhost	10.200.50.28	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
28	Host_10.200.50.29	modhost	10.200.50.29	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
29	Host_10.200.50.30	modhost	10.200.50.30	255.255.255.255	black				Created_by_Object_Filler_v2.2	web
30	Host_10.200.50.31	modhost	10.200.50.31	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
31	Host_10.200.50.32	modhost	10.200.50.32	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
32	Host_10.200.50.33	modhost	10.200.50.33	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
33	Host_10.200.50.34	modhost	10.200.50.34	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
34	Host_10.200.50.35	modhost	10.200.50.35	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
35	Host_10.200.50.36	modhost	10.200.50.36	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
36	Host_10.200.50.37	modhost	10.200.50.37	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
37	Host_10.200.50.38	modhost	10.200.50.38	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
38	Host_10.200.50.39	modhost	10.200.50.39	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
39	Host_10.200.50.40	modhost	10.200.50.40	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail
40	Host_10.200.50.41	modhost	10.200.50.41	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
41	Host_10.200.50.42	modhost	10.200.50.42	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
42	Host_10.200.50.43	modhost	10.200.50.43	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
43	Host_10.200.50.44	modhost	10.200.50.44	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
44	Host_10.200.50.45	modhost	10.200.50.45	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
45	Host_10.200.50.46	modhost	10.200.50.46	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
46	Host_10.200.50.47	modhost	10.200.50.47	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail
47	Host_10.200.50.48	modhost	10.200.50.48	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail

Just for fun, set the line 41 so the Web and DNS properties have as their Protecting Gateway the current firewall we have defined. In my case the name of the gateway is *ngxr60a* – do it as appropriate in your case – the name is case sensitive and the gateway **must** exist before, otherwise an error while importing the objects with DBedit.

So, the line 41 should look like this



	A	B	C	D	E	F	G	H	I	J	K	L
40	Host_10.200.50.40	modhost	10.200.50.40	255.255.255.255	black				Created_by_Object_Filler_v2.2	dns;mail		
41	Host_10.200.50.41	modhost	10.200.50.41	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail	ngxr60a	ngxr60a
42	Host_10.200.50.42	modhost	10.200.50.42	255.255.255.255	black				Created_by_Object_Filler_v2.2	web;dns;mail		

Save the changes, and exit from your spreadsheet program.

Now, we need to run Object Filler over the file we just created with the changes:

```
D:\Stuff\OFiller\v2.1>ofiller -f servers_only.csv -i csv -o setsrvprop.dbedit
Unofficial/Unsupported Object Filler V2.2  -  Developed by Martin Hoz
(c) 2003-2005 by Check Point Software Technologies, Inc.
===========================================================================
Processing objects...
...............
===========================================================================
It took 3 seconds of total processing time on QUIET Mode.
Processed 100 possible objects and/or rules.
Found 50 total valid (or successfully processed) objects/rules.
```

```
--------------------------------------------------------------------------------
Total successfully processed Hosts = 50
 - Total successfully processed Web Server Hosts = 20
   WARNING: These objects are only valid on NG+AI R55 or higher
 - Total successfully processed DNS Server Hosts = 30
   WARNING: These objects are only valid on NG+AI R55W or higher
 - Total successfully processed Mail Server Hosts = 30
   WARNING: These objects are only valid on NG+AI R55W or higher
--------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
================================================================================
Task done successfully! - Thank you for using Object Filler V2.2!
```

Well, it seems that all went fine. As expected, Object Filler warns about what versions support the given servers properties.

As well, the sum of the objects reported is more than 50, because some objects count for more than just one server type.

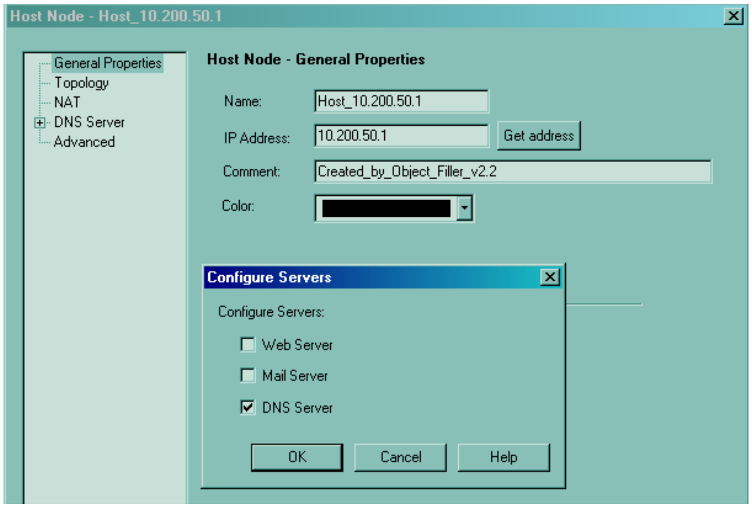
Let’s go through the process of importing the DBedit file into the SmartCenter. By now, you should know why each step in the process is needed:

```
[Expert@ngxr60]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> prompt
Interactive mode off.
ftp> get setsrvprop.dbedit
local: setsrvprop.dbedit remote: setsrvprop.dbedit
227 Entering passive mode (10,20,30,76,13,142)
125 Using existing data connection
#####################################
226 Closing data connection; File transfer successful.
37332 bytes received in 0.0253 secs (1.4e+03 Kbytes/sec)
ftp> bye
221 Service closing control connection
[Expert@ngxr60]# ls -la setsrvprop.dbedit
-rw-rw----    1 root     root        36852 Dec 26 15:50 setsrvprop.dbedit
[Expert@ngxr60]# cpstat mg

Product Name:  Check Point SmartCenter Server
Major version: 5
Minor version: 0
Build number:  593000414
```

```
Is started:    1
Active status: active
Status:        OK

Connected clients
----------------------------------------------
|Client type|Administrator|Host|Database lock|
----------------------------------------------
----------------------------------------------

[Expert@ngxr60]# dbedit -s localhost -u admin -f setsrvprop.dbedit
Enter Administrator Password: secret
Host_10.200.50.1 updated successfully.
Host_10.200.50.2 updated successfully.
   .
   .
   .
Host_10.200.50.49 updated successfully.
Host_10.200.50.50 updated successfully.
[Expert@ngxr60]#
```

That's it! – let's take a look at a couple of objects there to see if the modifications we wanted were actually done.

One of the DNS Servers:



Host Node - Host_10.200.50.1

Host Node - General Properties

Name: Host_10.200.50.1

IP Address: 10.200.50.1

Comment: Created_by_Object_Filler_v2.2

Configure Servers

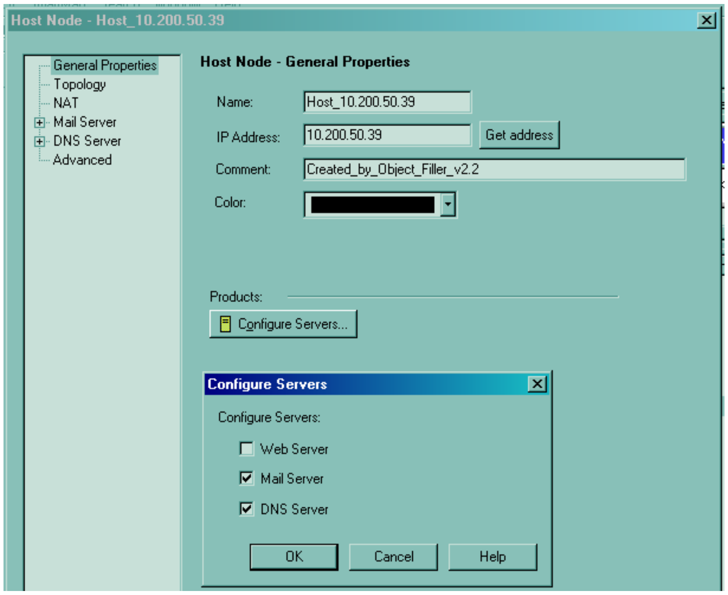
Configure Servers:

☐ Web Server

☐ Mail Server

☑ DNS Server

One of the Mail and DNS Servers:



Host Node - Host_10.200.50.39

Host Node - General Properties

Name: Host_10.200.50.39

IP Address: 10.200.50.39

Comment: Created_by_Object_Filler_v2.2

Color: ■

Configure Servers

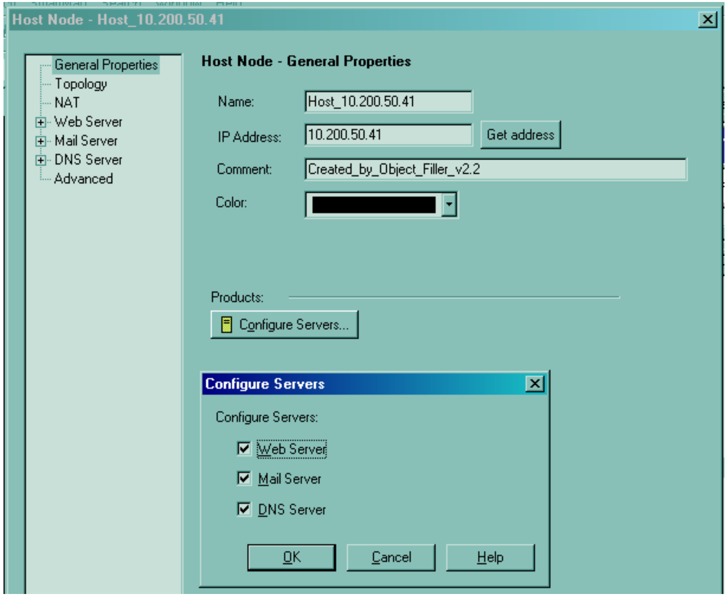
Configure Servers:

□ Web Server

☑ Mail Server

☑ DNS Server

One of the Web, Mail and DNS servers:



Host Node - Host_10.200.50.41

Host Node - General Properties

Name: Host_10.200.50.41

IP Address: 10.200.50.41

Comment: Created_by_Object_Filler_v2.2

Color: ■

Configure Servers

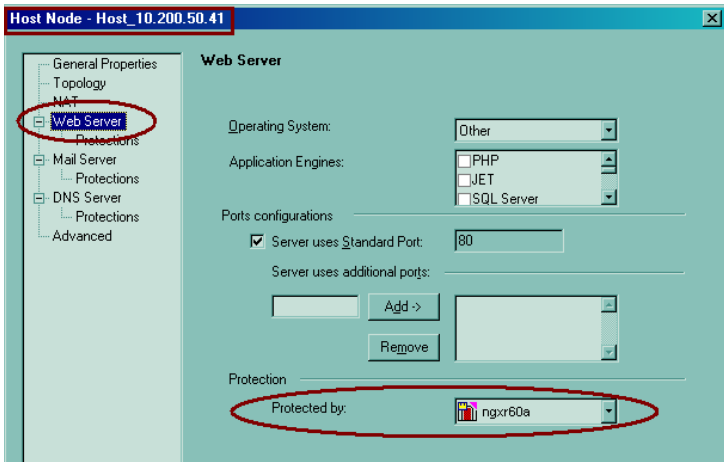
Configure Servers:

☑ Web Server

☑ Mail Server

☑ DNS Server

Let's see now how the Host_10.200.50.41 was set in the Protecting Gateway properties for DNS and Web protections:



Host Node - Host_10.200.50.41

Web Server

Operating System: Other

Application Engines: □ PHP □ JET □ SQL Server

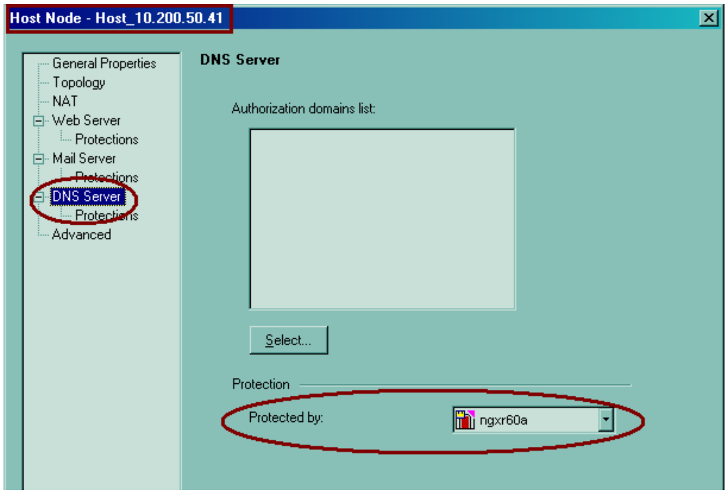
Ports configurations

☑ Server uses Standard Port: 80

Server uses additional ports:

Protection

Protected by: ngxr60a



Host Node - Host_10.200.50.41

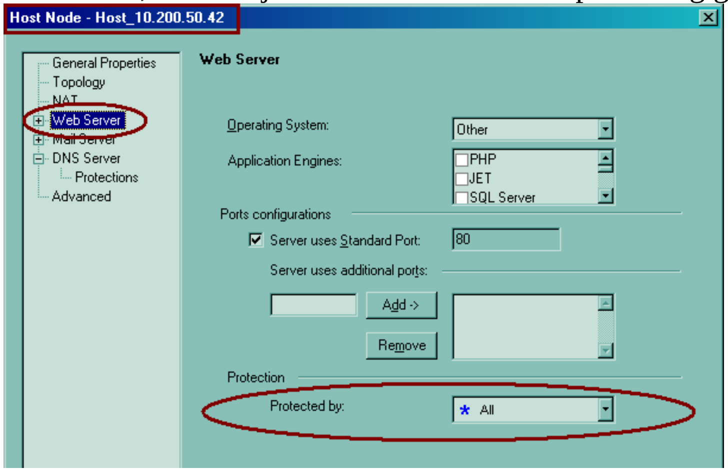
DNS Server

Authorization domains list:

Protection

Protected by: ngxr60a

In contrast, other objects have “All” as their protecting gateway:



Host Node - Host_10.200.50.42

Web Server

Operating System: Other

Application Engines: □ PHP □ JET □ SQL Server

Ports configurations

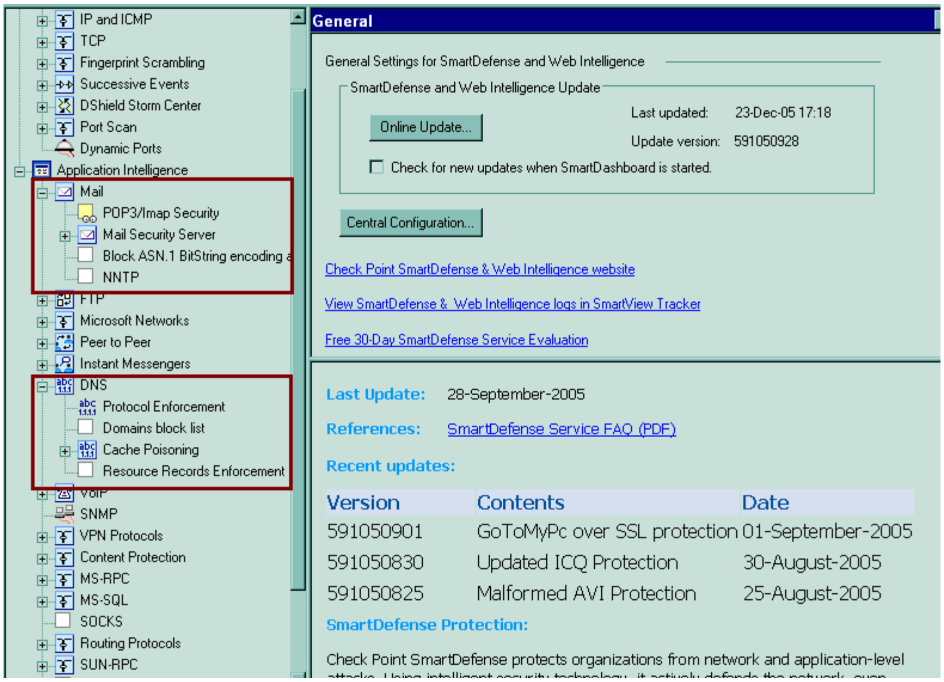
☑ Server uses Standard Port: 80

Server uses additional ports:

Protection

Protected by: * All

So, everything went as expected. The objects have now the Server properties needed in each case. So, now the last part is to tune in as well the SmartDefense properties for such things:



- IP and ICMP
- TCP
- Fingerprint Scrambling
- Successive Events
- DShield Storm Center
- Port Scan
- Dynamic Ports
- Application Intelligence
  - Mail
    - POP3/Imap Security
    - Mail Security Server
    - ☐ Block ASN.1 BitString encoding a
    - ☐ NNTP
  - FTP
  - Microsoft Networks
  - Peer to Peer
  - Instant Messengers
  - DNS
    - Protocol Enforcement
    - ☐ Domains block list
    - Cache Poisoning
    - ☐ Resource Records Enforcement
  - VoIP
  - SNMP
  - VPN Protocols
  - Content Protection
  - MS-RPC
  - MS-SQL
  - ☐ SOCKS
  - Routing Protocols
  - SUN-RPC

General

General Settings for SmartDefense and Web Intelligence

SmartDefense and Web Intelligence Update

Last updated: 23-Dec-05 17:18

Update version: 591050928

☐ Check for new updates when SmartDashboard is started.

Check Point SmartDefense & Web Intelligence website

View SmartDefense & Web Intelligence logs in SmartView Tracker

Free 30-Day SmartDefense Service Evaluation

Last Update: 28-September-2005

References: SmartDefense Service FAQ (PDF)

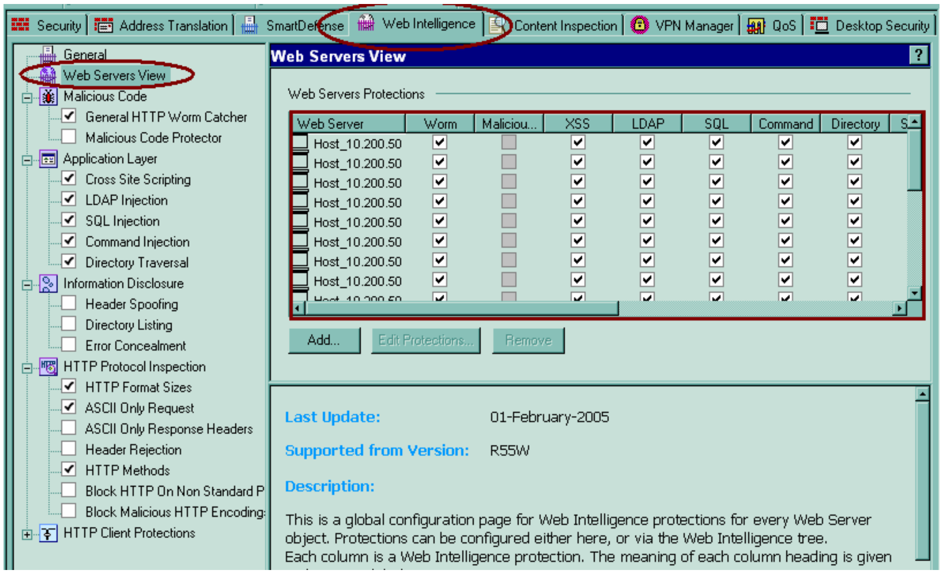
Recent updates:

Version	Contents	Date
591050901	GoToMyPc over SSL protection	01-September-2005
591050830	Updated ICQ Protection	30-August-2005
591050825	Malformed AVI Protection	25-August-2005

SmartDefense Protection:

Check Point SmartDefense protects organizations from network and application-level

You may as well tune the Web Intelligence settings for both the Server objects themselves and the at the Gateway level, using the Web Intellicence Tab:



- General
- Web Servers View
- Malicious Code
  - ☑ General HTTP Worm Catcher
  - ☐ Malicious Code Protector
- Application Layer
  - ☑ Cross Site Scripting
  - ☑ LDAP Injection
  - ☑ SQL Injection
  - ☑ Command Injection
  - ☑ Directory Traversal
- Information Disclosure
  - ☐ Header Spoofing
  - ☐ Directory Listing
  - ☐ Error Concealment
- HTTP Protocol Inspection
  - ☑ HTTP Format Sizes
  - ☑ ASCII Only Request
  - ☐ ASCII Only Response Headers
  - ☐ Header Rejection
  - ☑ HTTP Methods
  - ☐ Block HTTP On Non Standard P
  - ☐ Block Malicious HTTP Encoding
- HTTP Client Protections

Web Servers View

Web Servers Protections

Web Server	Worm	Maliciou...	XSS	LDAP	SQL	Command	Directory	S
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	
Host_10.200.50	☑	☐	☑	☑	☑	☑	☑	

Last Update: 01-February-2005

Supported from Version: R55W

Description:

This is a global configuration page for Web Intelligence protections for every Web Server object. Protections can be configured either here, or via the Web Intelligence tree.
Each column is a Web Intelligence protection. The meaning of each column heading is given

However, explaining what you need to tune and how, goes way beyond the purpose of the present document. You have very good documentation in the Products Documentation, the

SmartDashboard help (Hey! - F1 is your friend! ;-) – and as well in several online resources. So, use that to take the maximum advantage of your protection infrastructure… ☺

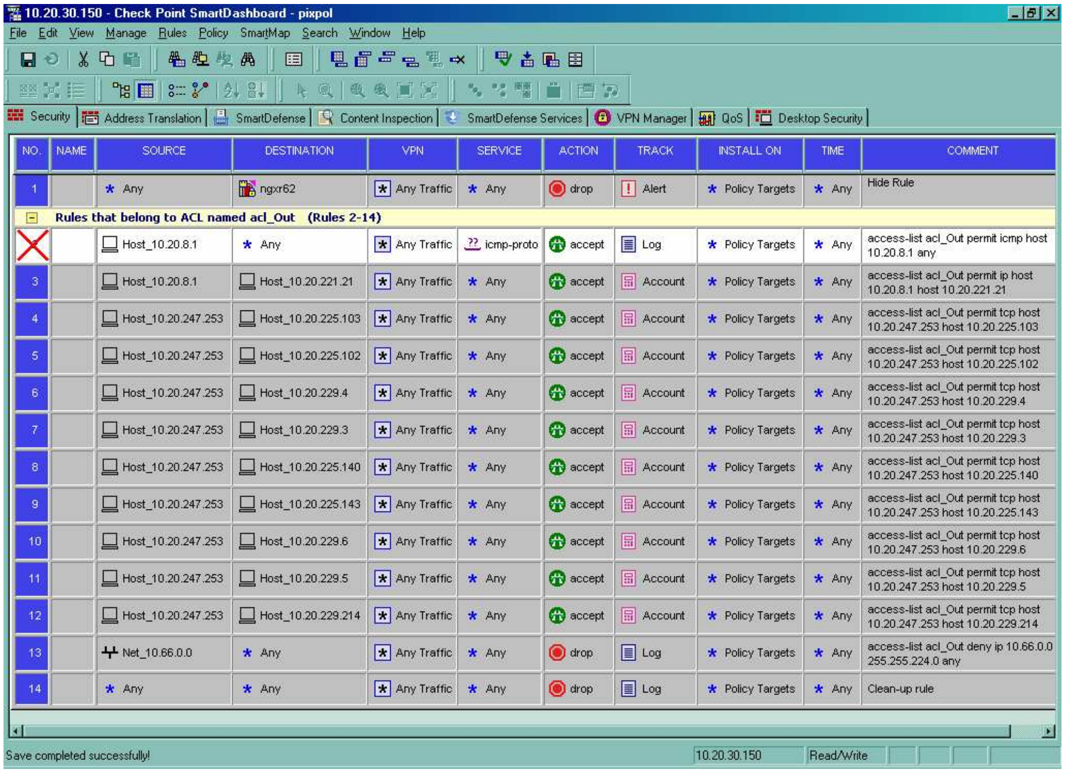
8. Special (strange or non-common) operations with rules

As of Object Filler and Object Dumper version 2.4, security rules are fully supported. The following sections show some of the most useful operations that can be eased by using Object Filler and Object Dumper.

First, let's assume you are running a SmartCenter with the configuration imported from the Cisco device on the steps 5.3 and 5.4 of this document only with rules from 1 to 12 and the addition of the clean-up and hide rules, as shown below.

8.1 Exporting rules from one place (SmartCenter or CMA) to a CSV File

Let's assume you have the following rulebase running on your SmartCenter Server


10.20.30.150 - Check Point SmartDashboard - pixpol
File Edit View Manage Rules Policy SmartMap Search Window Help
Security | Address Translation | SmartDefense | Content Inspection | SmartDefense Services | VPN Manager | QoS | Desktop Security

NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1		Any	ngxr62	Any Traffic	Any	drop	Alert	Policy Targets	Any	Hide Rule
Rules that belong to ACL named acl_Out (Rules 2-14)										
		Host_10.20.8.1	Any	Any Traffic	icmp-proto	accept	Log	Policy Targets	Any	access-list acl_Out permit icmp host 10.20.8.1 any
3		Host_10.20.8.1	Host_10.20.221.21	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.221.21
4		Host_10.20.247.253	Host_10.20.225.103	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.103
5		Host_10.20.247.253	Host_10.20.225.102	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.102
6		Host_10.20.247.253	Host_10.20.229.4	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.4
7		Host_10.20.247.253	Host_10.20.229.3	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.3
8		Host_10.20.247.253	Host_10.20.225.140	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.140
9		Host_10.20.247.253	Host_10.20.225.143	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.143
10		Host_10.20.247.253	Host_10.20.229.6	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.6
11		Host_10.20.247.253	Host_10.20.229.5	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.5
12		Host_10.20.247.253	Host_10.20.229.214	Any Traffic	Any	accept	Account	Policy Targets	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.214
13		Net_10.66.0.0	Any	Any Traffic	Any	drop	Log	Policy Targets	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
14		Any	Any	Any Traffic	Any	drop	Log	Policy Targets	Any	Clean-up rule

Save completed successfully!　10.20.30.150　Read/Write

Notice the added rules and the negated rule 2
You want to export those rules to a CSV File. You may do so with Object Dumper.

First, as you probably guess, we need to transfer the source file that holds all the rulebases, which is (precisely) *rulebases_5_0.fws* located under *$FWDIR/conf* – this file will be transferred from out SecurePlatform machine running the SmartCenter to our Windows machine where we have Object Dumper.

```
[Expert@ngxr62]# cd $FWDIR/conf
[Expert@ngxr62]# pwd
/opt/CPsuite-R62/fw1/conf
[Expert@ngxr62]# ls -la rulebases_5_0.fws
-rw-rw----    1 root     root        35825 Dec 15 16:07 rulebases_5_0.fws
[Expert@ngxr62]# file rulebases_5_0.fws
rulebases_5_0.fws: ASCII text
[Expert@ngxr62]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> prompt
Interactive mode off.
ftp> put rulebases_5_0.fws
local: rulebases_5_0.fws remote: rulebases_5_0.fws
227 Entering passive mode (10,20,30,76,10,131)
125 Using existing data connection
#####################################
226 Closing data connection; File transfer successful.
37456 bytes sent in 0.0666 secs (5.5e+02 Kbytes/sec)
```

Okay, now we have locally the file. Let’s use Object Dumper to dump the rulebases.

```
D:\Stuff\OFiller\v2.4>odumper -p rulebases_5_0.fws -o rules.csv
Unofficial/Unsupported Object Dumper v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================

===============================================================================
 * Processing rules...
-------------------------------------------------------------------------------
..................

===============================================================================
Processed 1631 possible objects and found 16 valid ones.
It took 2.0 seconds on quiet mode.
Total successfully processed Rules = 16
===============================================================================
Task done successfully! - Thank you for using Object Dumper v2.4!
```

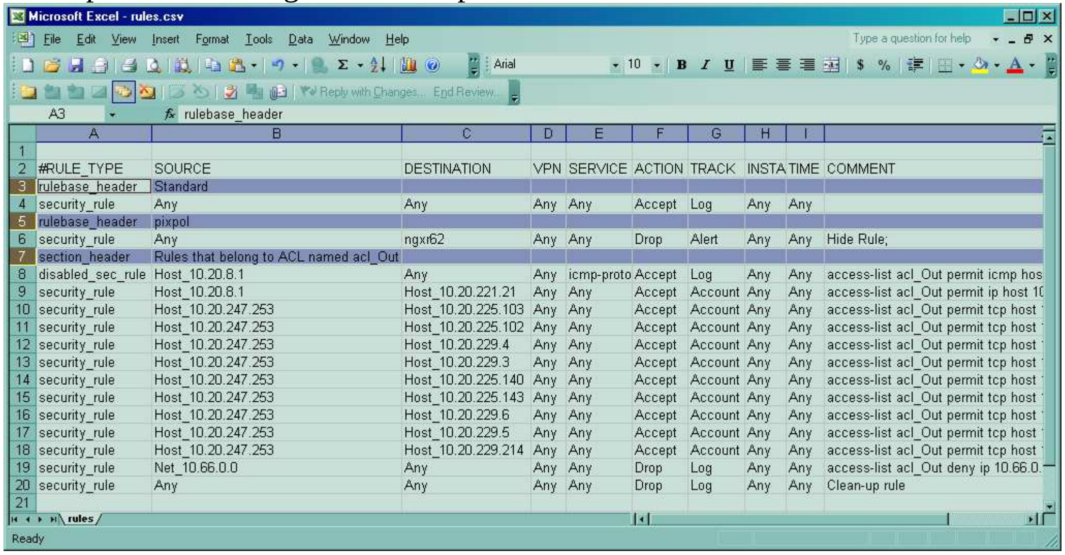
Please notice a few things:
- We used the *–p* switch to tell Object Dumper to process rules. We didn’t need to specify the *–f* switch since we were not processing any objects, however; if we want to export the objects as

well as the rules, you can specify the *–f* switch pointing to the *Objects_5_0.C* file and the *–p* switch pointing to the *rulebases_5_0.fws* file and all would be Ok.

- Please notice that we didn't use the *–v* (verbose) switch. We could of use it if we wanted more details as the lines were processed. It is left as an exercise to the reader to see how does this work using the *–v* switch, which is especially useful when something *strange* is *dumped* from the original configuration… - If you use the –v switch, redirect the output to a file and then open such file with a text editor. The syntax would be something like **`odumper -p rulebases_5_0.fws -o rules.csv –v > output.txt`** and then open the **`output.txt`** file using vi or notepad, for example.

- The number of rules reported is slightly different from what we expected (14). Let's see why...

Let's open the resulting file with a spreadsheet.


Microsoft Excel - rules.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
A3
rulebase_header
rules
Ready

	A	B	C	D	E	F	G	H	I	
1										
2	#RULE_TYPE	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTA	TIME	COMMENT
3	rulebase_header	Standard								
4	security_rule	Any	Any	Any	Any	Accept	Log	Any	Any	
5	rulebase_header	pixpol								
6	security_rule	Any	ngxr62	Any	Any	Drop	Alert	Any	Any	Hide Rule;
7	section_header	Rules that belong to ACL named acl_Out								
8	disabled_sec_rule	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	Any	Any	access-list acl_Out permit icmp hos
9	security_rule	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit ip host 10
10	security_rule	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
11	security_rule	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
12	security_rule	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
13	security_rule	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
14	security_rule	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
15	security_rule	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
16	security_rule	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
17	security_rule	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
18	security_rule	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host
19	security_rule	Net_10.66.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 10.66.0.
20	security_rule	Any	Any	Any	Any	Drop	Log	Any	Any	Clean-up rule
21										

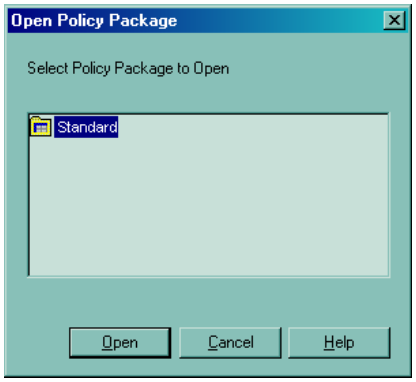
As usual, some things here deserve an explanation:

a) The type of rule is always shown at the beginning of each line. Possible values are *disabled_sec_rule* (means the rule shown is disabled), *security_rule* (means the following is a valid and enabled Security Rule), *section_header* (means it's the title for a section of rules, Also Known As *Section Title*) and *rulebase_header* (it's the name of the policy package, but you can see it as the name of the rule base or the policy name)

b) As of Object Filler and Object Dumper 2.4, rule names (which is a feature available on NGX R60 and up) are still NOT supported. While this is easy to fix and implement on the, it was left intentionally this way to make transitions from previous versions easier to new users of the tools.

c) You see on this case there are 2 policies. One named *Standard*, which is always the default name for the first policy; and a second policy named *pixpol*, which actually holds all the rules that are useful for us on this exercise.

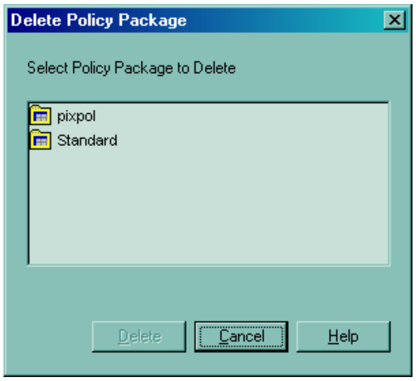
Typically you will see listed here several policies, not just two. Those are the policies you see when you are about to open a Policy Package from SmartDashboard (File, Open):



Open Policy Package

Select Policy Package to Open

The currently opened policy doesn't show up on this list. If you want to see all the policy packages then use Object Dumper as previously discussed, or use the *File*, *Delete*, *Entire Policy Package* option from *SmartDashboard*. If you are brave enough to use this option just to see the list of available policy packages, then be extra-careful on not deleting accidentally any policy there…



Delete Policy Package

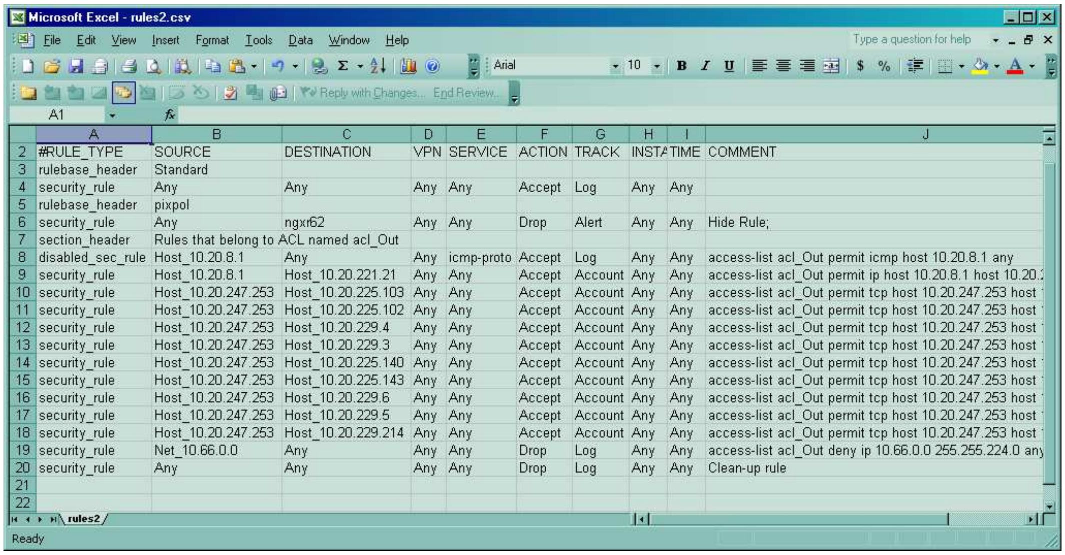
Select Policy Package to Delete

8.2 Modifying a fields on a exported (CSV) policy (rulebase) to get it imported back later…

Let's say that, on the policy exported, you need to do some modifications. In order to make it simple we'll do a basic modification, but the idea is the same if you want to modify anything on the rules.

Let's say that you want to modify the *Install on* column and make the rule apply only on the *ngxr62* gateway, thus we will change such column from *Any* to *ngxr62* on the CSV file.

The original file looks like this:

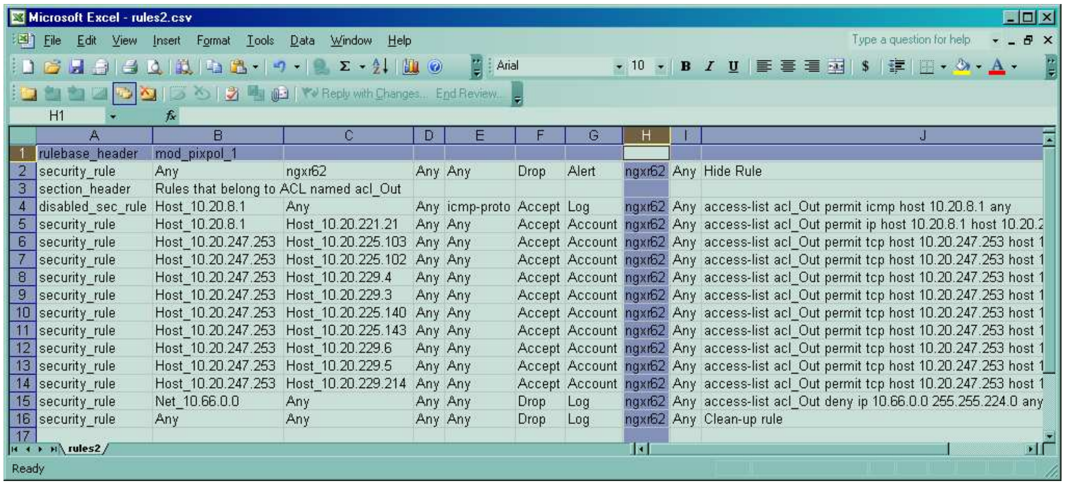

Microsoft Excel - rules2.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
A1
rules2
Ready

	A	B	C	D	E	F	G	H	I	J
2	#RULE_TYPE	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTA	TIME	COMMENT
3	rulebase_header	Standard								
4	security_rule	Any	Any	Any	Any	Accept	Log	Any	Any	
5	rulebase_header	pixpol								
6	security_rule	Any	ngxr62	Any	Any	Drop	Alert	Any	Any	Hide Rule;
7	section_header	Rules that belong to ACL named acl_Out								
8	disabled_sec_rule	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	Any	Any	access-list acl_Out permit icmp host 10.20.8.1 any
9	security_rule	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.
10	security_rule	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
11	security_rule	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
12	security_rule	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
13	security_rule	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
14	security_rule	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
15	security_rule	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
16	security_rule	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
17	security_rule	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
18	security_rule	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Account	Any	Any	access-list acl_Out permit tcp host 10.20.247.253 host
19	security_rule	Net_10.66.0.0	Any	Any	Any	Drop	Log	Any	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
20	security_rule	Any	Any	Any	Any	Drop	Log	Any	Any	Clean-up rule
21										
22										

Step 1: Strip the unnecessary lines so you only keep the policy (rulebase) you are interested on

Step 2: Change the policy name. This is because Object Filler is not able to make changes over current policies, BUT you can create a different policy and put the changes there. This is more or less equivalent to use the *Save As* option in the SmartDashboard. On this case, we wil change from *pixpol* to *mod_pixpol_1*.

At the end, the file should look like this:


Microsoft Excel - rules2.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
H1
rules2
Ready

	A	B	C	D	E	F	G	H	I	J
1	rulebase_header	mod_pixpol_1								
2	security_rule	Any	ngxr62	Any	Any	Drop	Alert	ngxr62	Any	Hide Rule
3	section_header	Rules that belong to ACL named acl_Out								
4	disabled_sec_rule	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	ngxr62	Any	access-list acl_Out permit icmp host 10.20.8.1 any
5	security_rule	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.2
6	security_rule	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
7	security_rule	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
8	security_rule	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
9	security_rule	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
10	security_rule	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
11	security_rule	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
12	security_rule	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
13	security_rule	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
14	security_rule	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
15	security_rule	Net_10.66.0.0	Any	Any	Any	Drop	Log	ngxr62	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
16	security_rule	Any	Any	Any	Any	Drop	Log	ngxr62	Any	Clean-up rule
17										

Now that all the changes are done, use Object Filler over the recently modified CSV file. Assuming the CSV file is named rules2.csv, you may use the following syntax:

```
D:\Stuff\OFiller\v2.4>ofiller -f rules2.csv -i csv -p policy -nopv -o rules2.dbedit
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
```

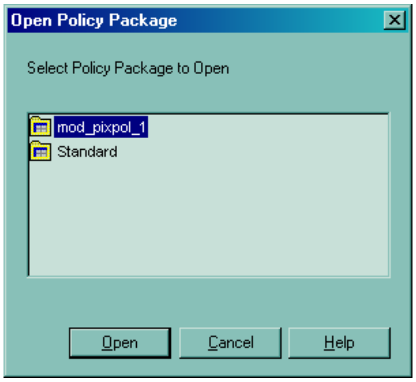
```
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================
Processing objects...

===============================================================================
Rules processing requested: Processing rules now!!!
............................................
===============================================================================
It took 3.0 seconds of total processing time on QUIET Mode.
Processed 46 possible objects and/or rules.
Found 14 total valid (or successfully processed) objects/rules.
Rules processing was requested and done for Policy "mod_pixpol_1".
-------------------------------------------------------------------------------
Total successfully processed Rules = 15
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

Some explanations:
- The –*p* switch (including the *policy* argument to the –*p* switch) is necessary to tell Object Filler to process rules. If you don’t use this switch Object Filler won’t process rules.
- The –*nopv* switch is necessary because otherwise Object Filler would check the objects referenced in the rules were processed by this Object Filler run. Since this is not the case, as you are processing only rules, then the –*nopv* switch is a must.

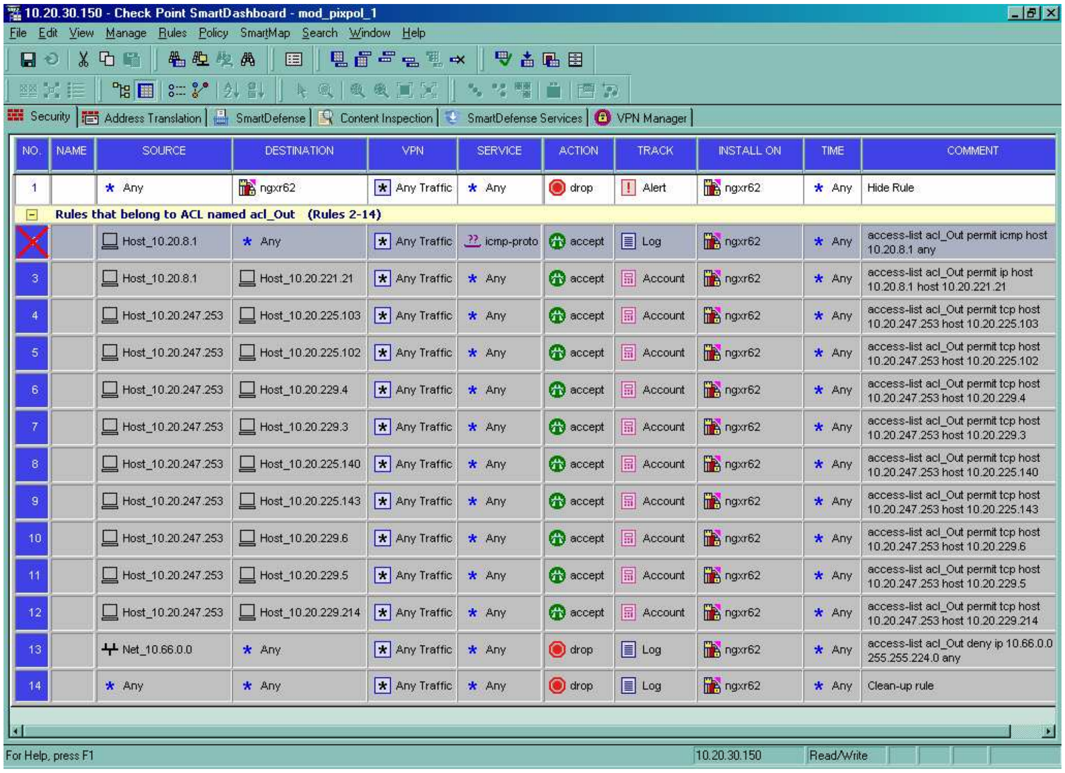
Now, use DBedit to enter the produced script into the SmartCenter. You should use the following syntax and then you will see the following result:

```
[Expert@ngxr62]# dbedit -s localhost -u admin -f rules2.dbedit
Enter Administrator Password: secret
mod_pixpol_1 updated successfully.
##mod_pixpol_1 updated successfully.
##mod_pixpol_1 updated successfully.
     .
     .
     .
##mod_pixpol_1 updated successfully.
##mod_pixpol_1 updated successfully.
```

Now let’s take a look at the SmartDashboard and see what the policy looks like. Use *File*, *Open* and then choose *mod_pixpol_1* from the available options



Now you may see the modified column shows what it was needed: Now the rules are installed over the *ngxr62* gateway



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	COMMENT
1		Any	ngxr62	Any Traffic	Any	drop	Alert	ngxr62	Any	Hide Rule
⊟ Rules that belong to ACL named acl_Out (Rules 2-14)										
		Host_10.20.8.1	Any	Any Traffic	icmp-proto	accept	Log	ngxr62	Any	access-list acl_Out permit icmp host 10.20.8.1 any
3		Host_10.20.8.1	Host_10.20.221.21	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.221.21
4		Host_10.20.247.253	Host_10.20.225.103	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.103
5		Host_10.20.247.253	Host_10.20.225.102	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.102
6		Host_10.20.247.253	Host_10.20.229.4	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.4
7		Host_10.20.247.253	Host_10.20.229.3	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.3
8		Host_10.20.247.253	Host_10.20.225.140	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.140
9		Host_10.20.247.253	Host_10.20.225.143	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.225.143
10		Host_10.20.247.253	Host_10.20.229.6	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.6
11		Host_10.20.247.253	Host_10.20.229.5	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.5
12		Host_10.20.247.253	Host_10.20.229.214	Any Traffic	Any	accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 10.20.229.214
13		Net_10.66.0.0	Any	Any Traffic	Any	drop	Log	ngxr62	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
14		Any	Any	Any Traffic	Any	drop	Log	ngxr62	Any	Clean-up rule

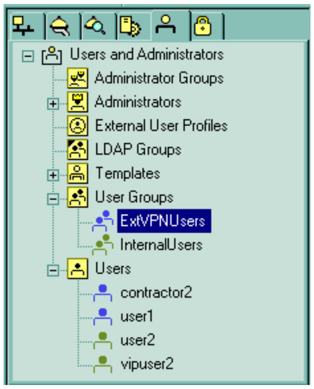
As you might guess, even though now our objective was to put *ngxr62* as the target of the rules (*Install On* column); you could remove elements from such column or any other as well (useful when you need to remove objects), use *search and replace* over the CSV file to modify the name or references to an specific element, or… use your imagination! ☺ - As long as you keep the syntax, the new referenced objects already exist in the objects database and you give a new name to the policy (rulebase) so it doesn't conflict with an existing one; all should be fine…

Mission accomplished! (by now). Let's get into harder tasks then…

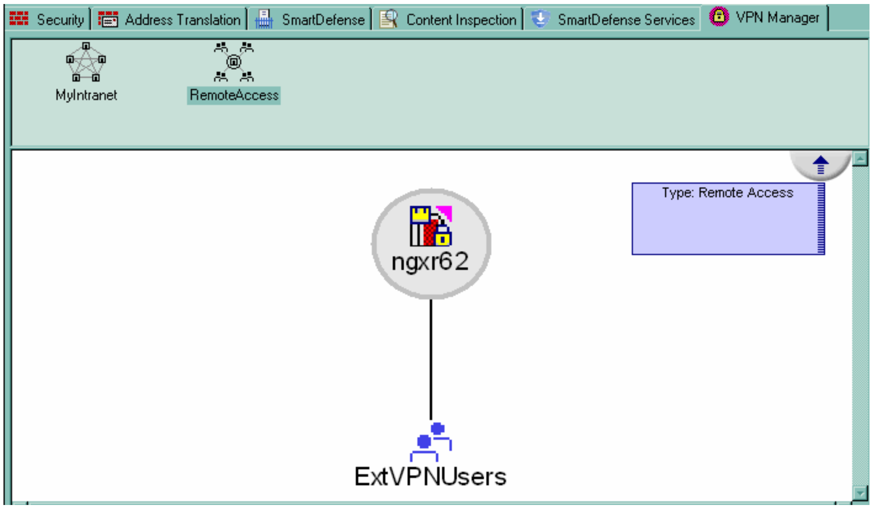
8.3 Adding rules to an existing policy (rulebase)

Let's say we want to add to the policy used on the section 8.2 two rules:
1 - Source: *ExtVPNUsers@Any*, Destination: *Net_10.66.0.0*, VPN: *RemoteAccess*, Service: *Any*, Action: *Accept*; Track: *Log*, Install On: *ngxr62*, Time: *Any*.
2 - Source: *InternalUsers@Net_10.66.0.0*, Destination: *Any*, VPN: *Any*, Service: *telnet;http*, Action: *UserAuth*; Track: *Log*, Install On: *ngxr62*, Time: *Any*.

These rules assume the User groups *ExtVPNUsers* and *InternalUsers* already exist:

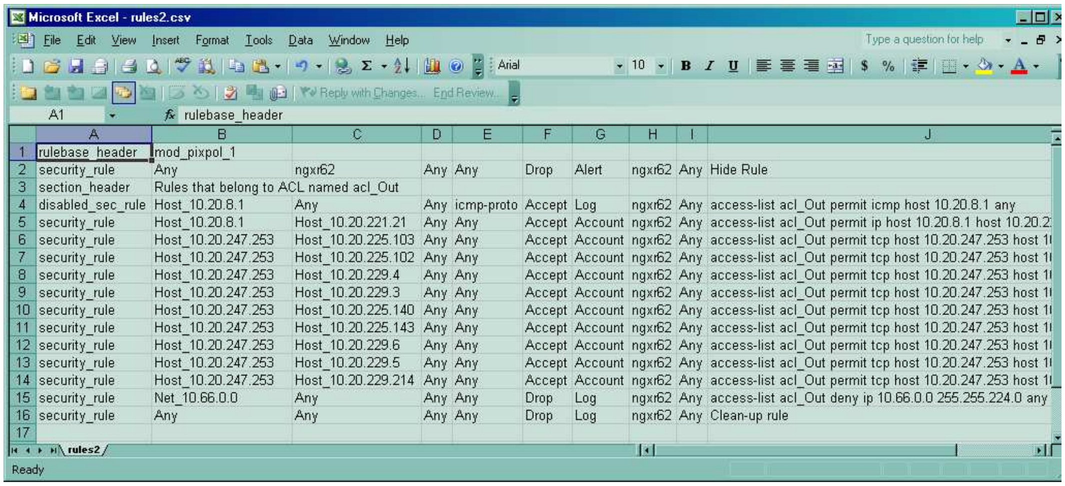


As well is assumes the *RemoteAccess* VPN community is properly configured, since one of the rules will involve the use of it



Let's remember that since Object Filler can't modify existing policies (rulebases), we need to change the policy (rulebase) name. On this case, we'll change it from *mod_pixpol_1* to *mod_pixpol_2*.

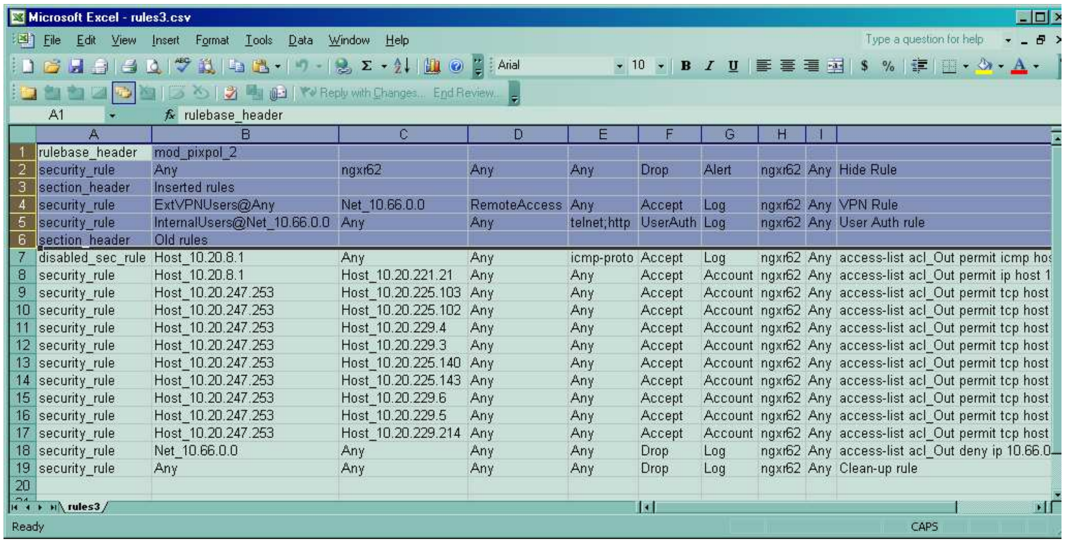
Okay. Let's get our hands dirty. The original rulebase looks like this:



Microsoft Excel - rules2.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
A1
rulebase_header
rules2
Ready

	A	B	C	D	E	F	G	H	I	J
1	rulebase_header	mod_pixpol_1								
2	security_rule	Any	ngxr62	Any	Any	Drop	Alert	ngxr62	Any	Hide Rule
3	section_header	Rules that belong to ACL named acl_Out								
4	disabled_sec_rule	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	ngxr62	Any	access-list acl_Out permit icmp host 10.20.8.1 any
5	security_rule	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit ip host 10.20.8.1 host 10.20.2
6	security_rule	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
7	security_rule	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
8	security_rule	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
9	security_rule	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
10	security_rule	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
11	security_rule	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
12	security_rule	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
13	security_rule	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
14	security_rule	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host 10.20.247.253 host 1
15	security_rule	Net_10.66.0.0	Any	Any	Any	Drop	Log	ngxr62	Any	access-list acl_Out deny ip 10.66.0.0 255.255.224.0 any
16	security_rule	Any	Any	Any	Any	Drop	Log	ngxr62	Any	Clean-up rule
17										

After the modifications, the file looks like this (the changed lines are highlighted):



Microsoft Excel - rules3.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
A1
rulebase_header
rules3
Ready
CAPS

	A	B	C	D	E	F	G	H	I	
1	rulebase_header	mod_pixpol_2								
2	security_rule	Any	ngxr62	Any	Any	Drop	Alert	ngxr62	Any	Hide Rule
3	section_header	Inserted rules								
4	security_rule	ExtVPNUsers@Any	Net_10.66.0.0	RemoteAccess	Any	Accept	Log	ngxr62	Any	VPN Rule
5	security_rule	InternalUsers@Net_10.66.0.0	Any	Any	telnet;http	UserAuth	Log	ngxr62	Any	User Auth rule
6	section_header	Old rules								
7	disabled_sec_rule	Host_10.20.8.1	Any	Any	icmp-proto	Accept	Log	ngxr62	Any	access-list acl_Out permit icmp hos
8	security_rule	Host_10.20.8.1	Host_10.20.221.21	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit ip host 1
9	security_rule	Host_10.20.247.253	Host_10.20.225.103	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
10	security_rule	Host_10.20.247.253	Host_10.20.225.102	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
11	security_rule	Host_10.20.247.253	Host_10.20.229.4	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
12	security_rule	Host_10.20.247.253	Host_10.20.229.3	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
13	security_rule	Host_10.20.247.253	Host_10.20.225.140	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
14	security_rule	Host_10.20.247.253	Host_10.20.225.143	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
15	security_rule	Host_10.20.247.253	Host_10.20.229.6	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
16	security_rule	Host_10.20.247.253	Host_10.20.229.5	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
17	security_rule	Host_10.20.247.253	Host_10.20.229.214	Any	Any	Accept	Account	ngxr62	Any	access-list acl_Out permit tcp host
18	security_rule	Net_10.66.0.0	Any	Any	Any	Drop	Log	ngxr62	Any	access-list acl_Out deny ip 10.66.0
19	security_rule	Any	Any	Any	Any	Drop	Log	ngxr62	Any	Clean-up rule
20										

Check out that we also modified the *section_header* name for the unmodified rules, and the new *section_header* used to put under it the added rules.

Now, let’s run Object Filler over the resulting file…

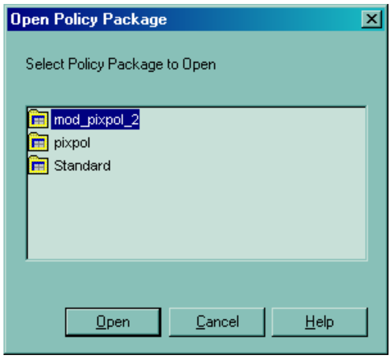
```
D:\Stuff\OFiller\v2.4>ofiller -f rules3.csv -i csv -nopv -p policy -o rules3.dbedit
Unofficial/Unsupported Object Filler v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
==============================================================================
Processing objects...

==============================================================================
Rules processing requested: Processing rules now!!!
.....................................................
==============================================================================
It took 3.0 seconds of total processing time on QUIET Mode.
```

```
Processed 54 possible objects and/or rules.
Found 16 total valid (or successfully processed) objects/rules.
Rules processing was requested and done for Policy "mod_pixpol_2".
-------------------------------------------------------------------------------
Total successfully processed Rules = 18
-------------------------------------------------------------------------------
Please review that all DBedit output commands were written correctly.
Please remember DBedit commands are imported into SmartCenter directly.
If you wish to review first, the use of CSV mode (-a switch) is suggested.
===============================================================================
Task done successfully! - Thank you for using Object Filler v2.4!
```

… and then import the results into SmartCenter using DBedit…

```
[Expert@ngxr62]# dbedit -s localhost -u admin -f rules3.dbedit
Enter Administrator Password: secret
mod_pixpol_2 updated successfully.
##mod_pixpol_2 updated successfully.
##mod_pixpol_2 updated successfully.
   .
   .
   .
##mod_pixpol_2 updated successfully.
##mod_pixpol_2 updated successfully.
```

… open the resulting policy using *File*, *Open* from SmartDashboard…

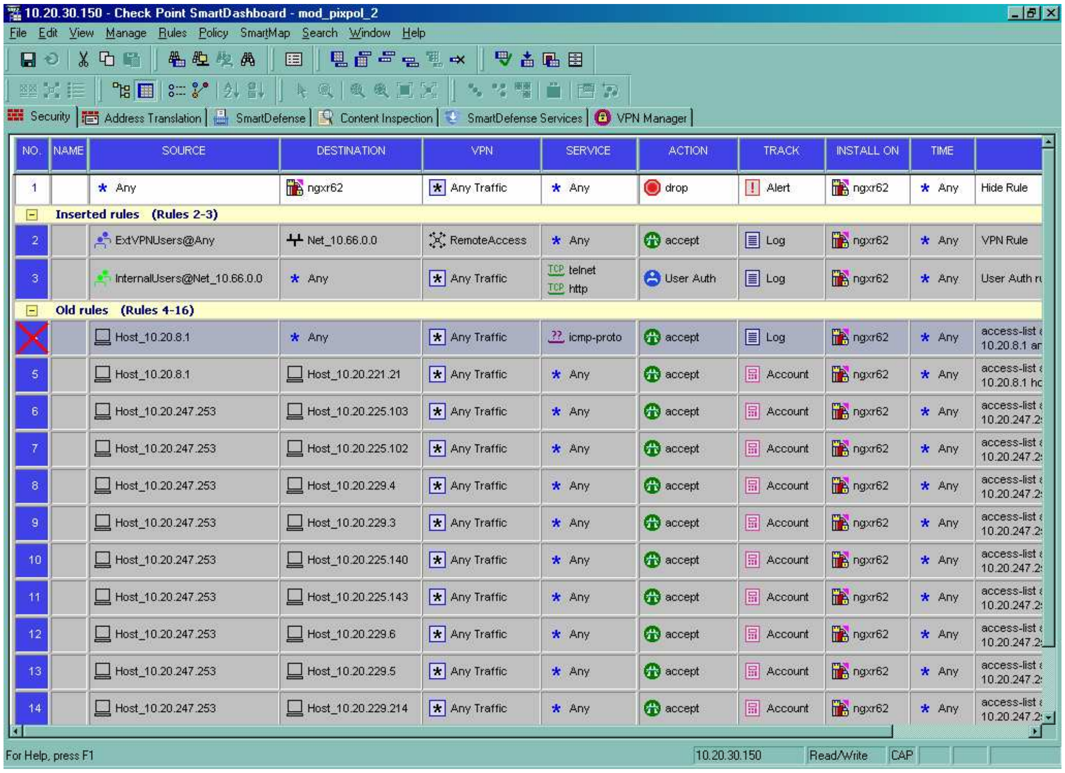


Open Policy Package

Select Policy Package to Open

- mod_pixpol_2
- pixpol
- Standard

… and finally check the results on SmartDashboard



NO.	NAME	SOURCE	DESTINATION	VPN	SERVICE	ACTION	TRACK	INSTALL ON	TIME	
1		* Any	ngxr62	Any Traffic	* Any	drop	Alert	ngxr62	* Any	Hide Rule
⊟ Inserted rules (Rules 2-3)										
2		ExtVPNUsers@Any	Net_10.66.0.0	RemoteAccess	* Any	accept	Log	ngxr62	* Any	VPN Rule
3		InternalUsers@Net_10.66.0.0	* Any	Any Traffic	TCP telnet TCP http	User Auth	Log	ngxr62	* Any	User Auth ru
⊟ Old rules (Rules 4-16)										
		Host_10.20.8.1	* Any	Any Traffic	icmp-proto	accept	Log	ngxr62	* Any	access-list 10.20.8.1 ar
5		Host_10.20.8.1	Host_10.20.221.21	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.8.1 hc
6		Host_10.20.247.253	Host_10.20.225.103	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
7		Host_10.20.247.253	Host_10.20.225.102	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
8		Host_10.20.247.253	Host_10.20.229.4	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
9		Host_10.20.247.253	Host_10.20.229.3	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
10		Host_10.20.247.253	Host_10.20.225.140	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
11		Host_10.20.247.253	Host_10.20.225.143	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
12		Host_10.20.247.253	Host_10.20.229.6	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
13		Host_10.20.247.253	Host_10.20.229.5	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2
14		Host_10.20.247.253	Host_10.20.229.214	Any Traffic	* Any	accept	Account	ngxr62	* Any	access-list 10.20.247.2

Seems the rules were created as needed! - Well done!

9. Recovering objects from a Gateway when SmartCenter has crashed and no backups are available

Scary: You have a distributed installation (SmartCenter separated from the gateways). You just for some reason didn’t back up your SmartCenter. You just walk in to the office one day and then you realize your SmartCenter Server is dead. No login, nothing. Then you realize the hard disk of your SmartCenter Server is dead and your resuscitation powers over SmartCenter Servers just expired yesterday! Real scary!!!

9.1 Recovering the objects

These are the good news: There is a file left on every Check Point Gateway, where the objects used to configure the policy installed on such gateway are stored.

The path to such file is $FWDIR/database/objects.C and you may run Object Dumper (2.2 or up) over it to get the network objects (including groups and services) listed on a CSV file. Once you have such objects listed there you may use Object Filler later to recreate them into your SmartCenter server.

Since we already taught you how to create objects from a CSV file using Object Filler to later import them into the SmartCenter Server using DBedit, we’ll focus only on dumping the list of objects used on the rules applied to the gateway.

First, let’s locate the file. If you are using something different from SecurePlatform for the gateway, the path to the file ($FWDIR/database/objects.C) is still valid:

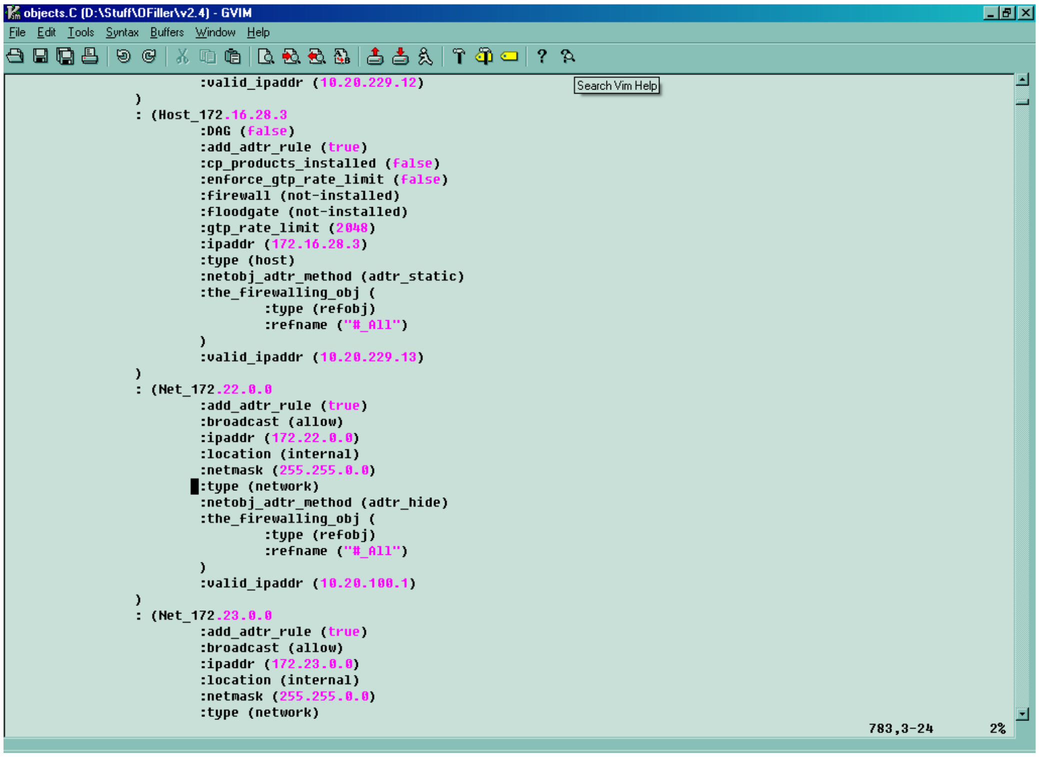
```
[Expert@ngxr62]# cd $FWDIR/database
[Expert@ngxr62]# pwd
/opt/CPsuite-R62/fw1/database
[Expert@ngxr62]# ls -la objects.C
-rw-rw----    1 root     root       843496 Dec 15 16:40 objects.C
 [Expert@ngxr62]# file objects.C
objects.C: ASCII text
```

Now let’s transfer it from our SecurePlatform to the Windows machine where Object Dumper is located.

```
[Expert@ngxr62]# ftp 10.20.30.76
Connected to 10.20.30.76 (10.20.30.76).
220 3Com 3CDaemon FTP Server Version 2.0
Name (10.20.30.76:admin): ofiller
331 User name ok, need password
Password: secret
230 User logged in
Remote system type is UNIX.
Using binary mode to transfer files.
```

```
ftp> ascii
200 Type set to A.
ftp> hash
Hash mark printing on (1024 bytes/hash mark).
ftp> prompt
Interactive mode off.
ftp> put objects.C
local: objects.C remote: objects.C
227 Entering passive mode (10,20,30,76,12,152)
125 Using existing data connection
######################################################################################
######################################################################################
######################################################################################
#######################################################
226 Closing data connection; File transfer successful.
879174 bytes sent in 0.782 secs (1.1e+03 Kbytes/sec)
ftp>
```

Now let’s take a fast look into the file


objects.C (D:\Stuff\OFiller\v2.4) - GVIM
File Edit Tools Syntax Buffers Window Help
Search Vim Help

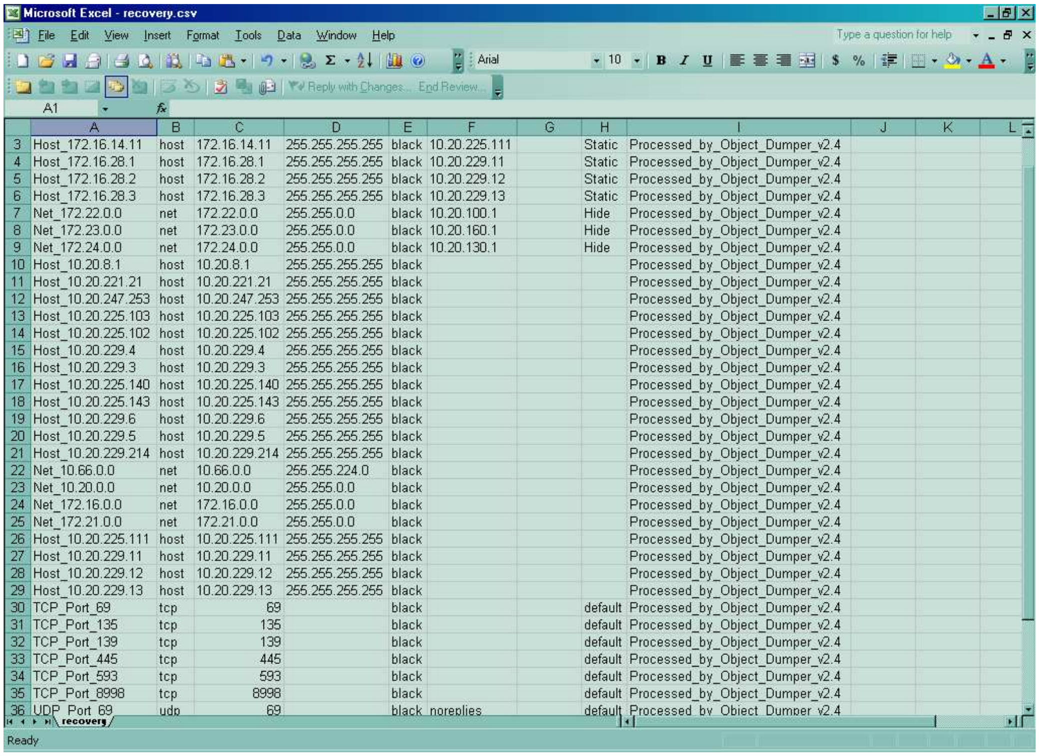
```
                        :valid_ipaddr (10.20.229.12)
                )
                : (Host_172.16.28.3
                        :DAG (false)
                        :add_adtr_rule (true)
                        :cp_products_installed (false)
                        :enforce_gtp_rate_limit (false)
                        :firewall (not-installed)
                        :floodgate (not-installed)
                        :gtp_rate_limit (2048)
                        :ipaddr (172.16.28.3)
                        :type (host)
                        :netobj_adtr_method (adtr_static)
                        :the_firewalling_obj (
                                :type (refobj)
                                :refname ("#_All")
                        )
                        :valid_ipaddr (10.20.229.13)
                )
                : (Net_172.22.0.0
                        :add_adtr_rule (true)
                        :broadcast (allow)
                        :ipaddr (172.22.0.0)
                        :location (internal)
                        :netmask (255.255.0.0)
                        :type (network)
                        :netobj_adtr_method (adtr_hide)
                        :the_firewalling_obj (
                                :type (refobj)
                                :refname ("#_All")
                        )
                        :valid_ipaddr (10.20.100.1)
                )
                : (Net_172.23.0.0
                        :add_adtr_rule (true)
                        :broadcast (allow)
                        :ipaddr (172.23.0.0)
                        :location (internal)
                        :netmask (255.255.0.0)
                        :type (network)
                                                                              783,3-24      2%
```

Okay. It’s very similar to the Objects_5_0.C file we used in other occasions, but the format of the file is different.

Let’s now use Object Dumper do see what can we get out of it.

```
D:\Stuff\OFiller\v2.4>odumper -f objects.C -o recovery.csv
Unofficial/Unsupported Object Dumper v2.4  -  Developed by Martin Hoz
(c) 2003-2006 by Check Point Software Technologies, Inc.
===============================================================================
===============================================================================
 * Processing objects...
-------------------------------------------------------------------------------
........................................

===============================================================================
Processed 35678 possible objects and found 40 valid ones.
From these, 7 were NATted records. It took 3.0 seconds on quiet mode.
Total successfully processed CP Gateways = 1
Total successfully processed Hosts = 20
Total successfully processed Networks = 7
Total successfully processed TCP Services = 6
Total successfully processed UDP Services = 6
===============================================================================
Task done successfully! - Thank you for using Object Dumper v2.4!
```

As you can see, several objects were recognized. Let's take a look at the CSV file generated:



Microsoft Excel - recovery.csv
File Edit View Insert Format Tools Data Window Help
Type a question for help
Arial
10
Reply with Changes... End Review...
A1
recovery
Ready

	A	B	C	D	E	F	G	H	I	J	K	L
3	Host_172.16.14.11	host	172.16.14.11	255.255.255.255	black	10.20.225.111		Static	Processed_by_Object_Dumper_v2.4			
4	Host_172.16.28.1	host	172.16.28.1	255.255.255.255	black	10.20.229.11		Static	Processed_by_Object_Dumper_v2.4			
5	Host_172.16.28.2	host	172.16.28.2	255.255.255.255	black	10.20.229.12		Static	Processed_by_Object_Dumper_v2.4			
6	Host_172.16.28.3	host	172.16.28.3	255.255.255.255	black	10.20.229.13		Static	Processed_by_Object_Dumper_v2.4			
7	Net_172.22.0.0	net	172.22.0.0	255.255.0.0	black	10.20.100.1		Hide	Processed_by_Object_Dumper_v2.4			
8	Net_172.23.0.0	net	172.23.0.0	255.255.0.0	black	10.20.160.1		Hide	Processed_by_Object_Dumper_v2.4			
9	Net_172.24.0.0	net	172.24.0.0	255.255.0.0	black	10.20.130.1		Hide	Processed_by_Object_Dumper_v2.4			
10	Host_10.20.8.1	host	10.20.8.1	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
11	Host_10.20.221.21	host	10.20.221.21	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
12	Host_10.20.247.253	host	10.20.247.253	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
13	Host_10.20.225.103	host	10.20.225.103	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
14	Host_10.20.225.102	host	10.20.225.102	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
15	Host_10.20.229.4	host	10.20.229.4	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
16	Host_10.20.229.3	host	10.20.229.3	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
17	Host_10.20.225.140	host	10.20.225.140	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
18	Host_10.20.225.143	host	10.20.225.143	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
19	Host_10.20.229.6	host	10.20.229.6	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
20	Host_10.20.229.5	host	10.20.229.5	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
21	Host_10.20.229.214	host	10.20.229.214	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
22	Net_10.66.0.0	net	10.66.0.0	255.255.224.0	black				Processed_by_Object_Dumper_v2.4			
23	Net_10.20.0.0	net	10.20.0.0	255.255.0.0	black				Processed_by_Object_Dumper_v2.4			
24	Net_172.16.0.0	net	172.16.0.0	255.255.0.0	black				Processed_by_Object_Dumper_v2.4			
25	Net_172.21.0.0	net	172.21.0.0	255.255.0.0	black				Processed_by_Object_Dumper_v2.4			
26	Host_10.20.225.111	host	10.20.225.111	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
27	Host_10.20.229.11	host	10.20.229.11	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
28	Host_10.20.229.12	host	10.20.229.12	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
29	Host_10.20.229.13	host	10.20.229.13	255.255.255.255	black				Processed_by_Object_Dumper_v2.4			
30	TCP_Port_69	tcp	69		black			default	Processed_by_Object_Dumper_v2.4			
31	TCP_Port_135	tcp	135		black			default	Processed_by_Object_Dumper_v2.4			
32	TCP_Port_139	tcp	139		black			default	Processed_by_Object_Dumper_v2.4			
33	TCP_Port_445	tcp	445		black			default	Processed_by_Object_Dumper_v2.4			
34	TCP_Port_593	tcp	593		black			default	Processed_by_Object_Dumper_v2.4			
35	TCP_Port_8998	tcp	8998		black			default	Processed_by_Object_Dumper_v2.4			
36	UDP_Port_69	udp	69		black	noreplies		default	Processed_by_Object_Dumper_v2.4			

Wonderful! –from here, if you followed the Object Filler part of this tutorial, you should know how to get this data converted into DBedit commands, and from there how to use them to create the objects in the SmartCenter Server.

9.2 Recovering the rules

These are the bad news: As of Object Filler and Object Dumper 2.4 you can’t (yet) recover the rules. However you can run the following script (and I have to thank Peter Phan for the idea of this script) and then have a more clear idea on what the installed rules on the gateway are.

All the script does is to parse the *rules.C* file located on *$FWDIR/database* on the gateway and show the currently installed policy.

Perform the following procedure on the gateway (assuming SecurePlatform here, but it works as well on Solaris or Nokia IPSO. You may figure it out for other Operating Systems):

```
[Expert@ngxr62]# cd $FWDIR/database
[Expert@ngxr62]# pwd
/opt/CPsuite-R62/fw1/database
[Expert@ngxr62]# ls -la rules.C
-rw-rw----    1 root     root        85668 Dec 15 16:40 rules.C
[Expert@ngxr62]# file rules.C
rules.C: ASCII text
[Expert@ngxr62]# sed "/:rules-adtr/,/^$/d" rules.C |\
> egrep ": |:action|:disabled|:global_location|:through|\
> :time|:track|:dst|:install|:services|:src" | more
```

The following would be the result in case of just one *Quad A* (Any, Any, Any, Accept) rule installed on the gateway:

```
                : (rule-1
                        :action (
                                : (accept
                                        :action ()
                        :disabled (false)
                        :global_location (middle)
                        :time (
                                : (Any
                        :track (
                                : Log
                        :dst (
                                : (Any
                        :install (
                                : (Any
                        :services (
                                : (Any
                        :src (
                                : (Any
                        :through (
                                : (ReferenceObject
```

The following would be the first 3 rules of a policy (on this example, the policy used on the section 8 of the present document). No rule 2 is shown, since it is disabled on such policy…

```
                : (rule-1
                        :action (
                                : (drop
                                        :action ()
                        :disabled (false)
                        :global_location (middle)
                        :time (
                                : (Any
                        :track (
                                : Alert
                        :dst (
                                : ngxr62
                        :install (
                                : (Any
                        :services (
                                : (Any
                        :src (
                                : (Any
                        :through (
                                : (ReferenceObject
                : (rule-3
                        :action (
                                : (accept
                                        :action ()
                        :disabled (false)
                        :global_location (middle)
                        :time (
                                : (Any
                        :track (
                                : Account
                        :dst (
                                : Host_10.20.221.21
                        :install (
                                : (Any
                        :services (
                                : (Any
                        :src (
                                : Host_10.20.8.1
                        :through (
                                : (ReferenceObject
                : (rule-4
                        :action (
                                : (accept
                                        :action ()
                        :disabled (false)
                        :global_location (middle)
                        :time (
                                : (Any
                        :track (
                                : Account
                        :dst (
                                : Host_10.20.225.103
                        :install (
                                : (Any
                        :services (
                                : (Any
                        :src (
                                : Host_10.20.247.253
                        :through (
                                : (ReferenceObject
```

As you can see, while this doesn’t provide an automatic recovery for the rulebase, showing the installed rules in such a format makes it easier to recover the policy by manually entering it again… (sorry!, having no backups nor Management HA has some price tag on it! :-P) – Now, if somebody can work on a script that leaves the text above in such a format that Object Filler can read it, then you would have the chance to use Object Filler to recreate the rules.

Remember that Object Filler takes only CSV format as a valid feed for security rules. Perhaps a scripting wizard is willing to take the challenge, solve the puzzle and do something on this regard. If you are such scripting wizard, please share with the Check Point community your work. Many will appreciate it and the world will be a bit better ☺

10. Last Comments

The idea of this Tutorial document was to give you a quick but efficient introduction to the use of the Object Filler and Object Dumper tools, so you can perform some tasks easier using the Command Line Interface; or in other cases perform procedures that otherwise would be impossible to do in an automated way, such as importing Objects from a PIX configuration.

Hopefully now that you are reading these lines, it is because you went successfully through all the exercises shown in the document and now you are a consummated wizard in the dark arts of bulk objects (and rules) creation, exporting and modification. **Congratulations**!

Any feedback (good or back) about this document is always welcomed and encouraged. Please (please, please… please) feel free to write to the e-mail address of the author, shown at the beginning of the document.

If you want to contribute to the project, there are ways to do so. They are shown in the FAQ of the User's Manual. Take a look at questions 1.11, 1.12, 1.13 and 2.6

If you want to go beyond, you have the Object Filler and Object Dumper User's Manual to go beyond this point, and try more complicated things (such as more complex rule sets, services, Edge devices, etc.) as well as other type of files supported as input for Object Filler. Good luck in your journey good admin!

Cheers!