

## Frosty the Snow Crash - Answers and Winners

We Have Winners!!

At long last, I've scoured through the amazing entries you guys sent for the Frosty the Snow Crash challenge... just in time for your snowmen to melt on a fine Spring day. I apologize for the delay, but I had to finish writing my new 6-day SANS course and work with Matt Carpenter to write the new "It Happened One Friday" challenge, which has consumed me for 80+ hours per week over the past several months. But, now it's time to announce our winners for the Frosty the Snow Crash challenge. The competition was fierce, as many of you submitted brilliant work. I was dazzled with your technical prowess and delighted with your creativity.

Please remember that judging was very difficult. I spent between ten and ninety minutes reading each response and carefully divining the best of the best. Besides the actual winning entries, those that received honorable mention are quite brilliant, and deserve kudos. Don't be disappointed if you got an honorable mention but didn't win. An honorable mention in a contest like this is a major accomplishment, given the really high quality of the many entries we received.

Remember Challenge Fans, as always, we'll award three prizes: One for the best technical answer, one for the most creative answer that is technically correct, and one awarded to a winner chosen randomly. This month's prize is my book, Malware: Fighting Malicious Code, which I authored with Lenny Zeltser.

--Ed Skoudis, Intelguardians

Author, Counter Hack Reloaded

[del.icio.us Slashdot It!](http://del.icio.us/SlashdotIt/)

[Discuss in Forums {mos\\_smf\\_discuss:Dec 07 - Frosty the Snow Crash }](#)

[www.chicagocon.com](http://www.chicagocon.com)  
Frosty the Snow Crash

Answers & Winners

By Ed Skoudis

## Frosty Winners - Finally!

For the answers to the challenge, I strongly encourage you to read the winning entries, which we link to below. They cover the answers so well that there is no need for me to provide detailed commentary. However, I did want to make a couple of points:

Why was Frosty's Thinkpad not properly detecting the temperature change? The pre-Lenovo brand-change Thinkpad used in this challenge was specifically chosen because WMI (and its WMIC command-line interface) cannot successfully pull information from its thermal sensors. Even with the patch described in the winning creative challenge, WMI still cannot interface with these thermal sensors appropriately. However, other third party tools outside of WMI and WMIC can, because they implement the proper calls into the kernel to pull the temperature. Was I expecting you to know this cold (pardon the pun)? No, but I wanted you to be aware of the limitations that WMI and WMIC have when doing fine-grained interaction with hardware such as the BIOS and ACPI (Advanced Configuration and Power Interface) settings. My whole point with this component of the challenge was that you can generally trust what WMI and WMIC tell you about your operating system, but all bets are off with what it tells you about the underlying hardware! Be careful, because if you trust what WMIC tells you about your hardware, you could melt.

Why did the evil magician formulate many of his WMIC commands to invoke a cmd.exe? Another lesson I was trying to illustrate with the challenge was the ability to circumvent some of the limitations of WMIC by using it to invoke a cmd.exe into which we pump a command. While I love WMIC with every fiber of my being, I do realize that it has some limitations. There are some things it cannot do, and there are some things that are really hard to make it do. But, in some cases, we can accomplish what we want with another non-WMIC command. For example, adding a user at the Windows command line and putting the user into a group are pretty easy with the "net user" and "net localgroup" commands entered at a local command prompt. Instead of floundering around to try to make WMIC do these, we can instead use WMIC as a vehicle to execute a cmd.exe on a target machine, and then have the cmd.exe run a local command on our behalf. Sometimes, that's the simpler way to go, and I use the technique all the time when I'm doing system administration or penetration testing. So, for example, to add a user named "merry" to the administrators group, we simply want to run the command "net localgroup administrators merry /add" on the target machine. We can easily do that by making WMIC invoke a cmd.exe /c followed by the command we want to run. The /c option in cmd.exe tells our command shell to run a command for us. Oh, and this technique helps us dodge the limitation of some Windows commands that can only run locally. The "net localgroup" command doesn't have an option by itself to run remotely. So, we use WMIC to make our chosen local command run remotely on the target box, via a cmd.exe. In the end, we've got a helpful construct --WMIC as the launcher of a cmd.exe on a target system, and then the cmd.exe to run some common and helpful local command on the target. You can see in the challenge that our attacker did this a lot.

What was the purpose of the maniacal screed by the evil magician? OK... so I'm not much of a poet or songwriter. However, I wanted to make an important point about Netcat backdoors that rely on UDP (and inject a little Windows FOR loop iteration in the challenge). When using Netcat in listener mode (-l) for UDP (-u) to create a backdoor shell (-e cmd.exe), Netcat only strips off the first character of each entered line and hands it to the shell. The entire text is sent across the network, but the listener only feeds the first character from each line to the shell. Why is this so? It's just the way Netcat's UDP functionality is implemented. Attackers use this "bug" to their advantage to fool administrators, who may not realize the actual command being entered, because they look at the whole lines, not just the first characters. It's a way bad guys can disguise what they're up to. And, they do use this technique. My good buddy George Bakos mentioned a recent attack to me against one of his own personal research honeypots where the technique was used. Many people who sent responses to the challenge somehow thought that Netcat itself or the garbage pumped into Netcat were causing the system to burn CPU cycles. Not really. Instead, if you look at the first character of each line sent to Netcat, it spells out a Windows FOR loop, as follows:

```
FOR /L %I IN (1,0,2) DO @ECHO MELT
```

This command, when executed at a cmd.exe, will run a counting loop (FOR /L) with an incrementer variable of %I (which we don't use in the loop later but still has to be specified for the command to work), counting by starting at 1, counting in steps of zero, and counting to 2 (1,0,2). That is, it'll count forever, until someone kills it. At each step through the loop, we'll write on Standard Output the word MELT (ECHO MELT), turning off display of commands using the @ symbol so we have prettier output.

So, there you have it. That's why I included the specific elements I chose in the challenge. I hope the challenge was fun and educational for you. Now, let's get on to our announcement of the WINNERS!

First Place, Technical  
Icetek

[Read the winning entry here!](#)

Icetek's answers were concise and well thought out, illustrating the experimental spirit showing how this person analyzed each element of the challenge, really trying out the commands to see what they were doing. The big point that tipped the victory into Icetek's hands was his mentioning of disconnecting the system from the network and potentially rebooting it to stop the melting. Excellent work, Icetek!

Honorable Mention

Steven Foust - Steven's work was incredible. He almost won first place with his excellent analysis and reasoning skills. He didn't win, though, because of his limited answer on the WMIC and BIOS interaction. Very, very close, Steven. Please enter our future challenges. Your work is very much appreciated, and you should be proud.

Nathaniel Hartley - Nathaniel's work was excellent. His answers were concise and well articulated. The only thing preventing him from winning was in the clean-up section, he didn't mention disconnecting or shutting down the box.

Timothy Perkins - Timothy's answer was quite solid, nailing almost every aspect of the challenge.

Dan Roberts - Dan's answers were awesome, but he specifically said he did not want to win, given his victory in a previous challenge. Stepping aside from the glory of a victory in these challenges and letting another person shine is quite impressive, Dan, and I thank you for that. You'd have likely won otherwise. But, the quality of your technical answer certainly deserves an honorable mention.

First Place, Creative  
Paul Tarter

[Read the winning entry here!](#)

Ohmigosh! This answer is awesome. It's in-depth narrative, technical details, and ingenuity are fantastic. Kudos to Paul for his excellent work. I urge you all to read this to gain insights into tools and technical analysis techniques for dealing with this kind of attack.

Honorable Mention

Andrew Laman - Andrew's answers is hilarious. It's very good technically, and made me chuckle several times. Great work, Andrew, and worthy of a very hearty Honorable Mention!

Random Draw Winner\*: Zoher Anis

Thanks for your great contributions, guys. I'm always honored when you submit an entry in our challenges.

And, don't forget our new challenge It Happened One Friday. It's live right now, and just waiting for your analysis and fine answers.

Thank you again-

--Ed Skoudis

Co-Founder, Intelguardians

SANS Instructor

\* Today's Random Number was brought to you by the fine people at Random.Org, your one stop shop for free and commercial random number generation on the World Wide Web.\*\*

\*\*No, Random.Org is not a sponsor. But, I used their free service to generate a random number to choose the Random Draw winner. Oh, and I really like the cool services that they offer on-line for free. So there.